

Análisis de Fourier

Notas de clase

Juan Diego Chang

Prefacio

Estas son unas notas del curso de Análisis de Fourier. Trato aquí de sintetizar lo que considero necesario para nivel de licenciatura sobre este tema. También vemos algunas aplicaciones para ingeniería de esta rama de las matemáticas, particularmente, vamos a buscar aplicar los conceptos a la manipulación de imágenes. Es por esto, que si bien tratamos de exponer las ideas en el caso unidimensional, siempre trataremos de extender los conceptos a casos de varias dimensiones.

Iniciamos nuestro recorrido por medio de las series de Fourier, cuál es su idea desde una perspectiva de álgebra lineal y luego estudiamos sus aplicaciones en el análisis de señales y resolución de ecuaciones diferenciales, siendo la primera a la que deberíamos darle mayor énfasis. Si deseamos manipular señales, primero debemos saber analizarlas. Es una buena introducción para fenómenos que son periódicos.

Luego, generalizamos este concepto, expandiendo nuestro estudio a funciones que ya no son periódicas y cómo podemos analizarlas. Dedicamos también una buena parte a tratar varios ejemplos de funciones que nos servirán más adelante para la manipulación de las señales.

Continuamos con una herramienta para la manipulación de imágenes que es la convolución. Tratamos de entender qué es lo que hace y vemos algunos ejemplos de cómo se calcula así como su interpretación. Iniciamos a entender las aplicaciones en señales de una dimensión que podrían ser ondas sonoras o señales eléctricas. Culminamos el capítulo con la generalización a varias dimensiones.

Por último, nuestra aplicación estelar, la manipulación de imágenes. Es también una forma de introducir el análisis de Fourier en su forma discreta, ya que las imágenes que tenemos en una computadora no son continuas. Veremos entonces cómo implementar la teoría desarrollada a lo largo del curso para implementar un programa de manipulación de imágenes.

En el apéndice, tenemos una lista de códigos realizados en Python para la realización de las gráficas que ilustran los conceptos de este curso, así como la implementación de los filtros vistos en el último capítulo.

Para poder seguir este documento, es necesario tener conceptos básicos de álgebra lineal, cálculo, variable compleja y cálculo vectorial.

Espero que disfruten el curso y sobre todo que podamos aprender mucho juntos.

Índice general

1. Series de Fourier	1
1.1. Repaso de álgebra lineal, descomposición en elementos de una base	1
1.2. Funciones periódicas y base de Fourier	3
1.3. Aspectos sobre períodos y frecuencias	6
1.4. No todo es color de rosa	8
1.5. Series de Fourier con exponenciales complejas	9
1.6. ¿Convergencia? Sí, con las condiciones de Dirichlet	11
1.7. ¿Cómo las podemos utilizar?	12
1.7.1. Ecuación de calor	13
1.7.2. Solución de la ecuación de calor en un anillo por medio de series de Fourier	14
1.8. Períodos, frecuencias y espectro	15
1.9. Generalización a más dimensiones	16
2. Transformada de Fourier	19
2.1. Retomando la idea de la serie de Fourier con un ejemplo y tomando el espectro	19
2.2. La transformada de Fourier	21
2.3. Algunos comentarios sobre la transformada de Fourier	22
2.4. Transformada inversa de Fourier	24
2.5. Propiedades generales	25
2.5.1. Dualidades	25
2.5.2. Invariantes de paridad y de conjugación compleja	27
2.5.3. Linealidad	27
2.5.4. Teorema de traslación	28
2.5.5. Teorema de cambio de escala o de similitud	28
2.5.6. La transformada de Fourier de una derivada	29
2.6. Más ejemplos	29
2.6.1. Función triángulo	29
2.6.2. Decaimiento exponencial	30
2.6.3. Distribución gaussiana	31
2.6.4. Función impulso o delta de Dirac	35
2.6.5. Función constante	36
2.6.6. Escalón unitario	37
2.7. Transformada de Fourier en varias dimensiones	38
2.7.1. Propiedades	40

Índice general

2.7.2.	Algunos ejemplos	41
2.7.2.1.	Función rectángulo en dos dimensiones	41
2.7.2.2.	Función gaussiana	42
2.7.3.	Coordenadas polares	43
2.7.3.1.	Funciones de Bessel	46
2.7.3.2.	Ejemplo: funciones circ y jinc	46
3. Convolución y filtros		49
3.1.	Motivación, definiciones y teorema de convolución	49
3.2.	Convoluciones y transformada inversa	51
3.3.	Interpretación de la convolución	52
3.4.	Propiedades de la convolución	54
3.5.	Aplicaciones con filtros	56
3.5.1.	Pasa bajo	57
3.5.2.	Filtros pasa banda	58
3.5.3.	Filtro pasa alto	59
3.5.4.	Filtro elimina banda	60
3.6.	Generalización a varias dimensiones	61
4. Aplicación: tratamiento de imágenes		63
4.1.	Imágenes digitales	63
4.2.	Primeras manipulaciones	63
4.2.1.	Escala de grises	64
4.2.2.	Blanco y negro	64
4.2.3.	Negativo	65
4.2.4.	Otras manipulaciones	66
4.3.	Transformada de Fourier discreta	68
4.4.	Transformada rápida de Fourier	69
4.4.1.	Transformada discreta de Fourier en una dimensión: otra perspectiva	69
4.4.1.1.	Notaciones	69
4.4.1.2.	Transformada de Fourier discreta para vectores	70
4.4.2.	Transformada rápida de Fourier en una dimensión	71
4.4.3.	Transformada rápida de Fourier en dos dimensiones	73
4.5.	Convolución en versión discreta	73
Bibliografía		74
A. Códigos para gráficas		77
A.1.	Capítulo 1	77
A.1.1.	Figura 1.1	77
A.1.2.	Figuras 1.2 y 1.3	79
A.1.3.	Figuras 1.4	81

Índice general

A.1.4. Figuras 1.5 y 1.6	83
A.1.5. Figura 1.7	85
A.2. Capítulo 2	86
A.2.1. Figuras 2.1, 2.2, 2.3 y 2.4	86
A.2.2. Figuras 2.5 y 2.6	89
A.2.3. Figuras 2.7 y 2.8	90
A.2.4. Figura 2.9	91
A.2.5. Figura 2.10	92
A.2.6. Figuras 2.11 y 2.12	93
A.2.7. Figuras 2.13 y 2.14	94
A.2.8. Figura 2.15	96
A.3. Capítulo 3	97
A.3.1. Figura 3.1	97
A.3.2. Figura 3.2	99
A.3.3. Figuras 3.4 y 3.5	101
A.3.4. Figuras 3.6 y 3.7	102
A.4. Capítulo 4	103
A.4.1. Figura 4.2	103
A.4.2. Figura 4.3	105
A.4.3. Figura 4.4	106
A.4.4. Figura 4.5	107
A.4.5. Figura 4.6	110
A.4.6. Figura 4.7	112

1. Series de Fourier

La utilidad del análisis de Fourier cubre prácticamente todas las áreas de la ingeniería como el diseño de circuitos, espectroscopías, cristalografía, tratamiento de señales o tratamiento de imágenes. Iniciamos nuestro tratamiento del análisis de Fourier por medio de las series de Fourier y tratar de comprender qué son desde un punto de vista físico y matemático, tratando de extender conceptos adquiridos ya en los cursos de física básica y de matemáticas básicas.

Vamos a basar la idea fundamental de estas series en la característica fundamental de las ondas: el principio de interferencia (o en matemáticas, el principio de superposición). Una ilustración de esta propiedad puede verse en la figura 1.1. Otra característica fundamental es que la mayoría de estas ondas son periódicas o las consideramos periódicas al momento de estudiarlas.

Podemos ver que la suma (la figura de abajo) en la figura 1.1 es más complicada que las otras figuras que son como ondas planas. La mayoría de señales que recibimos en aparatos eléctricos o que recibimos de la naturaleza (por ejemplo en un telescopio) u ondas sonoras son tan o incluso más complicadas que esta suma de dos. La idea es entonces saber cómo podemos descomponer estas señales. En los capítulos siguientes, estudiamos cómo transformarlas.

El hecho que se sumen y que no alteremos la periodicidad, nos hace pensar en el concepto de espacio vectorial. De hecho, las ondas forman un espacio vectorial, llamado espacio de Hilbert¹. Así pues, repasaremos las ideas fundamentales de álgebra lineal que nos servirán para el curso.

1.1. Repaso de álgebra lineal, descomposición en elementos de una base

Consideremos una base B en \mathbb{R}^3

$$B = \left\{ \vec{u}_1 = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}, \vec{u}_2 = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}, \vec{u}_3 = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \right\}.$$

¹Omitiremos aquí la definición de un espacio de Hilbert.

1. Series de Fourier

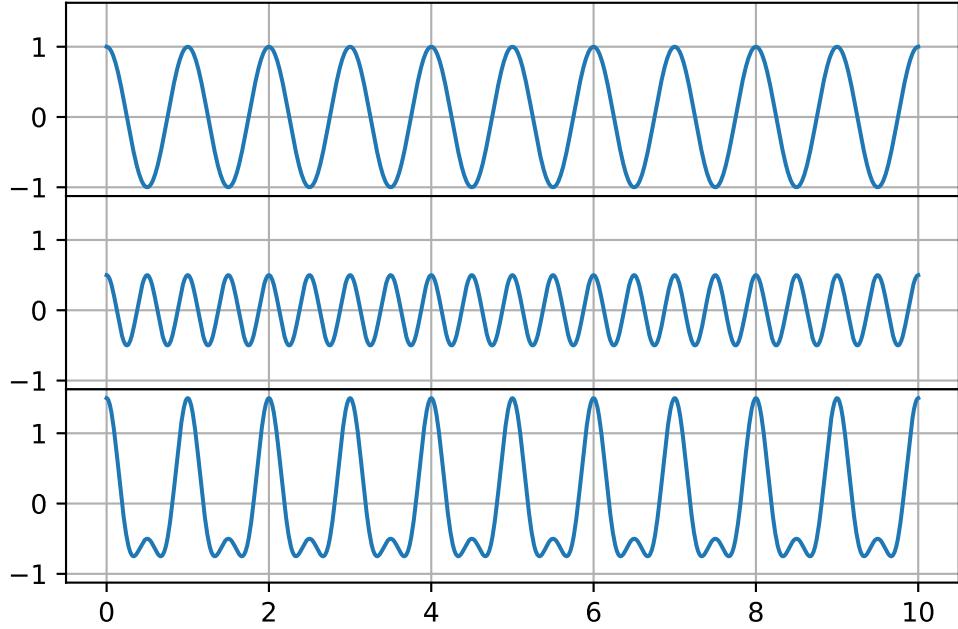


Figura 1.1.: Superposición de ondas. En la figura de arriba vemos $f(t) = \cos(2\pi t)$, en la del medio $g(t) = \frac{1}{2} \cos(4\pi t)$ y la de abajo, $h(t) = f(t) + g(t)$.

Ahora, si consideramos un vector $\vec{v} = \begin{pmatrix} 2 \\ 3 \\ 4 \end{pmatrix}$ puede descomponerse, en la base B , de la forma

$$\vec{v} = 2 \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} + 3 \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} = 2\vec{u}_1 + 3\vec{u}_2 + 4\vec{u}_3.$$

Con la base B esto es muy sencillo, pero ¿cómo podemos obtener de forma sistemática estos coeficientes? Esto no es muy complicado, gracias al producto escalar:

$$2 = \vec{u}_1 \cdot \vec{v} = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} \cdot \begin{pmatrix} 2 \\ 3 \\ 4 \end{pmatrix},$$

y así sucesivamente. Para tener finalmente,

$$\vec{v} = \vec{u}_1 \cdot \vec{v} + \vec{u}_2 \cdot \vec{v} + \vec{u}_3 \cdot \vec{v}.$$

Ahora, ¿cómo podemos hacer esto con funciones? Para esto, es necesario generalizar el concepto de producto escalar. Empecemos por generalizar esto a \mathbb{R}^n y consideremos a dos vectores $\vec{u}, \vec{v} \in \mathbb{R}^n$, tenemos

$$\vec{u} \cdot \vec{v} = \sum_{i=1}^n u_i v_i,$$

donde u_i y v_i son las componentes de los vectores \vec{u} y \vec{v} respectivamente. Ahora, si generalizamos esto a sumas infinitas con índices continuos, y así tenemos la correspondencia siguiente

$$\begin{aligned} u_i &\rightarrow u(x) \\ v_i &\rightarrow v(x) \\ \sum_i &\rightarrow \int dx \\ \vec{u} \cdot \vec{v} &\rightarrow \langle u | v \rangle, \end{aligned}$$

entonces

$$\langle u | v \rangle = \int_a^b u(x) v(x) dx. \quad (1.1)$$

Ahora surge la pregunta, ¿qué base tomamos? Para responder esto, es necesario definir qué es una función periódica.

1.2. Funciones periódicas y base de Fourier

Ya con estos conceptos básicos de álgebra lineal, vamos a extender esto a funciones periódicas.

Definición 1.1. Una función f es una función periódica de período T si y solo si

$$f(t) = f(t + T),$$

para todo valor de t , si T es el menor valor para lo que esto se cumple, ya que esto es válido para cualquier valor $n \in \mathbb{Z}$ tal que

$$f(t) = f(t + nT).$$

Proposición 1.1. Proponemos a $\{a \cos(n\omega_0 t), b \sin(n\omega_0 t)\}$ con $n \in \mathbb{N}$ donde $\omega_0 = \frac{2\pi}{T}$ que se conoce como frecuencia angular y T es el período. Estas funciones tienen como dominio $[-\frac{T}{2}, \frac{T}{2}]$.

1. Series de Fourier

Demostración. Para mostrar esto, nos bastará con demostrar que las funciones son ortogonales entre sí. Para esto utilizaremos la ecuación 1.1 y mostraremos que las funciones son ortogonales entre sí. Notemos que no es necesario calcular todos los casos ya que las funciones de la base tienen propiedades de paridad: $\cos(x)$ es par y $\sin(x)$ es impar:

$$\cos(x) = \cos(-x) \text{ y } \sin(x) = -\sin(-x),$$

y tomamos en cuenta que una función par por una impar es impar, así que las integrales de funciones impares se anularán, así pues $\int_{-T/2}^{T/2} \cos(n\omega_0 t) \sin(n\omega_0 t) dt = 0$, para todo m y n enteros.

Consideremos ahora el caso en que $m \neq n$,

$$\begin{aligned} \int_{-T/2}^{T/2} a \cos(n\omega_0 t) b \cos(m\omega_0 t) dt &= \frac{ab}{2} \int_{-T/2}^{T/2} \cos((m+n)\omega_0 t) + \cos((n-m)\omega_0 t) dt \\ \int_{-T/2}^{T/2} a \cos(n\omega_0 t) b \cos(m\omega_0 t) dt &= \frac{ab}{2} \left(\frac{\sin((m+n)\omega_0 t)}{(m+n)\omega_0} + \frac{\sin((n-m)\omega_0 t)}{(n-m)\omega_0} \right)_{-T/2}^{T/2} = 0, \end{aligned}$$

y ahora,

$$\begin{aligned} \int_{-T/2}^{T/2} a \sin(n\omega_0 t) b \sin(m\omega_0 t) dt &= \frac{ab}{2} \int_{-T/2}^{T/2} -\cos((m+n)\omega_0 t) + \cos((n-m)\omega_0 t) dt \\ \int_{-T/2}^{T/2} a \sin(n\omega_0 t) b \sin(m\omega_0 t) dt &= \frac{ab}{2} \left(-\frac{\sin((m+n)\omega_0 t)}{(m+n)\omega_0} + \frac{\sin((n-m)\omega_0 t)}{(n-m)\omega_0} \right)_{-T/2}^{T/2} = 0, \end{aligned}$$

Ahora si $m = n$,

$$\begin{aligned} \int_{-T/2}^{T/2} a^2 \cos^2(n\omega_0 t) dt &= \frac{a^2}{2} \int_{-T/2}^{T/2} 1 + \cos(2n\omega_0 t) dt \\ \int_{-T/2}^{T/2} a^2 \cos^2(n\omega_0 t) dt &= \frac{Ta^2}{2}, \end{aligned}$$

y ahora,

$$\begin{aligned} \int_{-T/2}^{T/2} a^2 \sin^2(n\omega_0 t) dt &= a^2 \int_{-T/2}^{T/2} 1 - \cos^2(n\omega_0 t) dt \\ \int_{-T/2}^{T/2} a^2 \sin^2(n\omega_0 t) dt &= a^2 \int_{-T/2}^{T/2} 1 - \frac{1}{2} - \cos(2n\omega_0 t) dt = \frac{Ta^2}{2}. \end{aligned}$$

Así pues, si queremos tener una base ortonormal (ortogonal y con vectores de norma 1, requerimos que

$$a = b = \sqrt{\frac{2}{T}}.$$

□

Al conjunto de $\{a \cos(n\omega_0 t), b \sin(n\omega_0 t)\}$, se le conoce como base de Fourier y a sus elementos como modos de vibración.

Así pues, tenemos que una función periódica de período T , definida en el intervalo $[-\frac{T}{2}, \frac{T}{2}]$ puede escribirse como una combinación lineal de estas funciones de la forma

$$f(t) = \frac{1}{2}a_0 + \sum_{n=1}^{\infty} a_n \cos(n\omega_0 t) + b_n \sin(n\omega_0 t), \quad (1.2)$$

tal que,

$$a_0 = \frac{2}{T} \int_{-T/2}^{T/2} f(t) dt, \quad a_n = \frac{2}{T} \int_{-T/2}^{T/2} f(t) \cos(n\omega_0 t) dt, \quad b_n = \frac{2}{T} \int_{-T/2}^{T/2} f(t) \sin(n\omega_0 t) dt. \quad (1.3)$$

A esta expresión se le conoce como la serie de Fourier de la función f , y a a_n, b_n como coeficientes de Fourier.

Proposición 1.2. Si $f(t)$ es par, entonces $b_n = 0$ para todo n . Si $f(t)$ es impar, entonces $a_n = 0$ para todo n .

Demostración. Recordamos que la integral del producto de una función par con una impar es nula, así pues evaluando los coeficientes de Fourier de la ecuación 1.3 son nulas con estas condiciones. \square

Ejemplo 1.1. Determinar la serie de Fourier de la función $f(t)$ definida por

$$f(t) = \begin{cases} -1 & -T/2 < t < 0 \\ 1 & 0 < t < T/2, \end{cases}$$

y es tal que $f(t) = f(t + T)$. Podemos ver su gráfica en la figura 1.2.

Nótese que f es impar, entonces sólo basta calcular b_n , de manera que

$$\begin{aligned} b_n &= \frac{2}{T} \int_{-T/2}^{T/2} f(t) \sin(n\omega_0 t) dt \\ &= \frac{2}{T} \left(\int_{-T/2}^0 -\sin(n\omega_0 t) dt + \int_0^{T/2} \sin(n\omega_0 t) dt \right) \\ &= \frac{2}{T} \left(\left(-\frac{\cos(n\omega_0 t)}{n\omega_0} \right) \Big|_{-T/2}^0 + \left(\frac{\cos(n\omega_0 t)}{n\omega_0} \right) \Big|_0^{T/2} \right) \\ &= \frac{2}{Tn\omega_0} (-1 + (-1)^n + (-1)^n - 1) \\ &= \frac{2}{n\pi} (-1 + (-1)^n) = \frac{2}{n\pi} ((-1)^n - 1) \\ b_n &= \begin{cases} 0 & \text{para } n \text{ par} \\ \frac{4}{n\pi} & \text{para } n \text{ impar,} \end{cases} \end{aligned}$$

entonces

$$f(t) = \frac{4}{\pi} \sum_{n=\text{impar}}^{\infty} \frac{1}{n} \sin(n\omega_0 t).$$

Veamos gráficamente cómo se ve esto en la figura 1.3, como mientras más términos agregamos a la serie más nos acercamos a la función original de la figura 1.2.

Queda claro con este ejemplo, que es posible aproximar una función por medio de una serie finita de Fourier. Denotamos a esta aproximación como $S_k(t)$, que es la serie con los primeros k términos de la serie, es decir

$$S_k(t) = \frac{a_0}{2} + \sum_{n=1}^k (a_n \cos(n\omega_0 t) + b_n \sin(n\omega_0 t)).$$

1. Series de Fourier

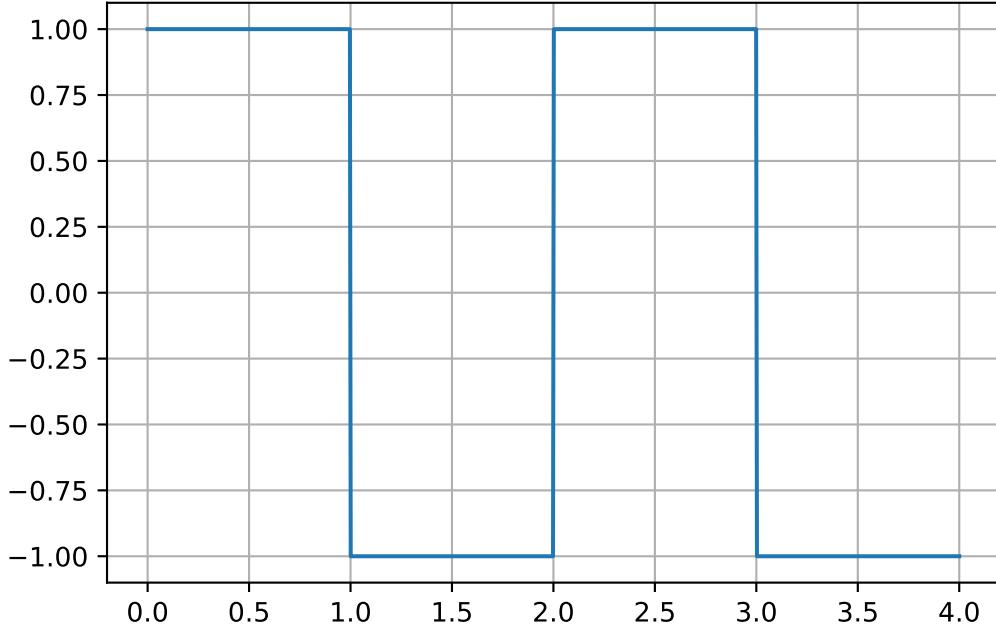


Figura 1.2.: Función del ejemplo 1.

No está de más notar que en las esquinas parece que la aproximación con 100 términos no es del todo buena, a este fenómeno se le conoce como fenómeno de Gibbs. Esto es porque estamos sumando funciones continuas y la suma finita de funciones continuas es continua, y la función $f(t)$ no es continua. Requerimos de más términos en la serie para ir removiendo estas anomalías.

1.3. Aspectos sobre períodos y frecuencias

Vamos a hacer una analogía aquí con conceptos físicos de movimientos periódicos, vistos en los cursos de física. Recordemos que para funciones de nuestra base de Fourier, digamos

$$A \sin(n\omega_0 t),$$

tenemos un período T que en este caso es tal que

$$\omega_0 = \frac{2\pi}{T},$$

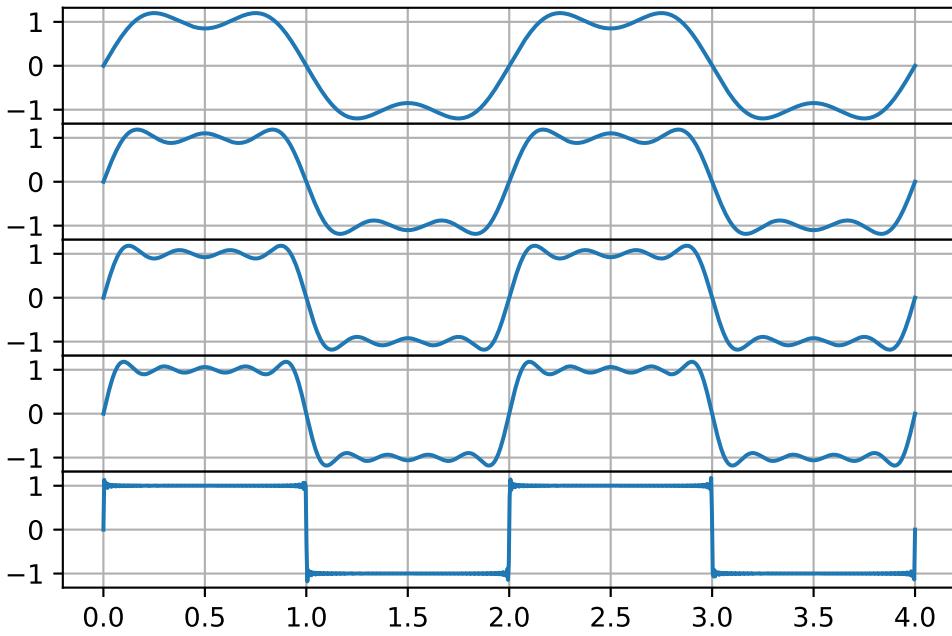


Figura 1.3.: Construcción de la serie de Fourier en el ejemplo 1. En la figura de arriba, tomamos los primeros dos términos de la serie, en la siguiente, 3 términos, luego, 4 términos, luego 5 términos y por último 100 términos. Notamos cómo cada vez nos acercamos a la figura original.

conocida como frecuencia angular, así pues, la frecuencia ν_0 de nuestra función es $\nu_0 = 1/T$, por lo que

$$\nu_0 = \frac{\omega_0}{2\pi}.$$

Son directamente proporcionales. Entonces, cuando tenemos una señal periódica y realizamos su descomposición en series de Fourier, los primeros términos de la serie son de baja frecuencia y entre más grandes sean los términos, vamos agregando modos de más alta frecuencia.

Entonces, en el ejemplo anterior, para poder aproximar de mejor manera las discontinuidades, es necesario utilizar términos de frecuencias altas.

1. Series de Fourier

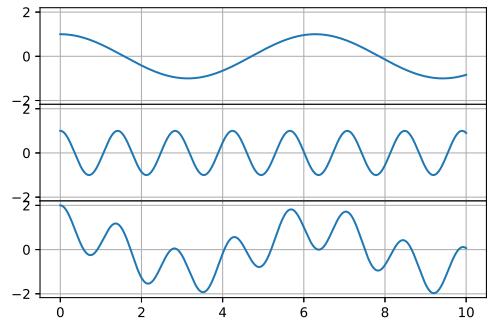
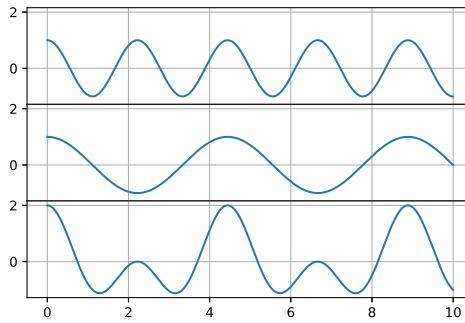


Figura 1.4.: La suma de funciones periódicas a veces es periódica y otras veces no. Del lado izquierdo vemos un ejemplo de funciones periódicas cuya suma es periódica y del lado derecho una que no.

1.4. No todo es color de rosa

Aunque no será de nuestro interés, vale la pena resaltar que la suma de dos funciones periódicas, no siempre es una función periódica. Requerimos de una situación particular y esta es que, por ejemplo, la función

$$f(t) = \cos(\omega_1 t) + \cos(\omega_2 t)$$

es periódica solamente si

$$\frac{\omega_1}{\omega_2} \in \mathbb{Q}, \quad (1.4)$$

así pues, si $\omega_1 = 1$ y $\omega_2 = \sqrt{2}$, tendremos una función que no es periódica al calcular $f(t)$, ahora si tomamos a $\omega_1 = 2\sqrt{2}$. Veamos esto en la figura 1.4.

Veamos un análisis más riguroso. Consideraremos entonces la función f y veamos bajo qué condiciones existe T tal que

$$f(t+T) = f(t).$$

Calculemos entonces $f(t+T)$

$$\begin{aligned} f(t+T) &= \cos(\omega_1 t + \omega_1 T) + \cos(\omega_2 t + \omega_2 T) \\ &= \cos(\omega_1 t) \cos(\omega_1 T) - \sin(\omega_1 t) \sin(\omega_1 T) + \cos(\omega_2 t) \cos(\omega_2 T) - \sin(\omega_2 t) \sin(\omega_2 T). \end{aligned}$$

Así pues necesitamos que se reúnan las condiciones

$$\begin{aligned} \cos(\omega_1 T) &= 1 \\ \sin(\omega_1 T) &= 0 \\ \cos(\omega_2 T) &= 1 \\ \sin(\omega_2 T) &= 0, \end{aligned}$$

1.5. Series de Fourier con exponentiales complejas

lo que se traduce en

$$\begin{aligned}\omega_1 T &= 2\pi n \\ \omega_2 T &= 2\pi m,\end{aligned}$$

con $n, m \in \mathbb{N}$. De manera que

$$\frac{\omega_1}{n} = \frac{\omega_2}{m} \Leftrightarrow \frac{m}{n} = \frac{\omega_2}{\omega_1},$$

recobrando así la condición 1.4. Entonces, ¿podemos considerar realmente esta como una base adecuada? El truco está en que ω_i es siempre un múltiplo entero para la base de Fourier. Sin embargo, es difícil realizar gráficas con números irracionales, así pues, una computadora siempre encontrará funciones que son periódicas y por eso le ponemos poca atención a esta restricción (no lo haríamos si fuéramos matemáticos). Así pues, si realizamos la gráfica de la figura 1.4 con un dominio suficientemente grande, encontraremos periodicidad.

1.5. Series de Fourier con exponentiales complejas

Resulta bastante útil, para efectos de calcular el uso de exponentiales complejas. Para hacer esto, recordemos que, partiendo de la identidad

$$e^{it} = \cos(t) + i \sin(t),$$

obtenemos,

$$\cos(t) = \frac{e^{it} + e^{-it}}{2} \quad \text{y} \quad \sin(t) = \frac{e^{it} - e^{-it}}{2i},$$

así pues, podemos ver cómo se expresa una serie de Fourier en términos exponentiales

$$\begin{aligned}f(t) &= \frac{1}{2}a_0 + \sum_{n=1}^{\infty} a_n \cos(n\omega_0 t) + b_n \sin(n\omega_0 t) \\ &= \frac{a_0}{2} + \sum_{n=1}^{\infty} a_n \frac{e^{in\omega_0 t} + e^{-in\omega_0 t}}{2} + b_n \frac{e^{in\omega_0 t} - e^{-in\omega_0 t}}{2i} \\ &= \frac{a_0}{2} + \sum_{n=1}^{\infty} e^{in\omega_0 t} \frac{(a_n - ib_n)}{2} + e^{-in\omega_0 t} \frac{(a_n + ib_n)}{2} \\ f(t) &= \sum_{n=-\infty}^{\infty} c_n e^{in\omega_0 t}.\end{aligned}\tag{1.5}$$

Notemos que aquí $n \in \mathbb{Z}$, es decir que puede tomar valores negativos. Para $n \geq 0$, tenemos que $c_n = a_n - ib_n$ (aquí, $b_0 = 0$) y para $n < 0$, $c_n = a_n + ib_n$. Ahora veámoslos explícitamente

$$\begin{aligned}\frac{a_n - ib_n}{2} &= \frac{1}{T} \left(\int_{-T/2}^{T/2} f(t) \cos(n\omega_0 t) dt - i \int_{-T/2}^{T/2} f(t) \sin(n\omega_0 t) dt \right) \\ &= \frac{1}{T} \int_{-T/2}^{T/2} f(t) \left(\frac{e^{in\omega_0 t} + e^{-in\omega_0 t}}{2} - \frac{e^{in\omega_0 t} - e^{-in\omega_0 t}}{2i} \right) dt \\ \frac{a_n - ib_n}{2} &\equiv \hat{f}(n).\end{aligned}\tag{1.6}$$

1. Series de Fourier

De igual manera,

$$\begin{aligned}\frac{a_n+ib_n}{2} &= \frac{1}{T} \left(\int_{-T/2}^{T/2} f(t) \cos(n\omega_0 t) dt + i \int_{-T/2}^{T/2} f(t) \sin(n\omega_0 t) dt \right) \\ &= \frac{1}{T} \int_{-T/2}^{T/2} f(t) \left(\frac{e^{in\omega_0 t} + e^{-in\omega_0 t}}{2} + \frac{e^{in\omega_0 t} - e^{-in\omega_0 t}}{2i} \right) dt \\ &= \frac{1}{T} \int_{-T/2}^{T/2} f(t) e^{in\omega_0 t} dt \\ \frac{a_n+ib_n}{2} &\equiv \hat{f}(n).\end{aligned}\tag{1.7}$$

Más adelante veremos el significado de estos coeficientes de Fourier y por qué usamos la notación $\hat{f}(n)$. En general, definimos entonces

$$\hat{f}(n) = \frac{1}{T} \int_{-T/2}^{T/2} f(t) e^{in\omega_0 t} dt\tag{1.8}$$

Vale la pena resaltar que aquí son los términos cercanos a $n = 0$ los de baja frecuencia y conforme va creciendo $|n|$, vamos agregando los términos de mayor frecuencia.

Analicemos otro ejemplo.

Ejemplo 1.2. Analicemos ahora la señal conocida como función sierra que es

$$f(t) = \begin{cases} t & 0 \leq t < 1 \\ 0 & \text{en cualquier otro caso.} \end{cases}$$

Resulta que esta función extendida tiene período 1, y su gráfica es la figura 1.5 .

Evaluemos ahora los coeficientes de Fourier de esta función, así pues

$$\hat{f}(n) = \int_0^1 t e^{-2\pi int} dt.$$

Si $n = 0$, tenemos simplemente que

$$\hat{f}(0) = 1,$$

y para $n \neq 0$, tenemos que, integrando por partes,

$$\hat{f}(n) = \left[-\frac{te^{-2\pi int}}{2\pi int} \right]_0^1 - \int_0^1 \frac{-1}{2\pi int} e^{-2\pi int} dt = \frac{i}{2\pi n}.$$

Por lo tanto, la serie de Fourier es

$$f(t) = \frac{1}{2} + \sum_{n=-\infty}^{\infty} \frac{i}{2\pi n} e^{2\pi int}$$

y por lo tanto tratemos de ver cómo se aproxima ahora las sumas parciales de Fourier a esta función. Esto puede verse en la figura 1.6.

De nuevo, tenemos el fenómeno de Gibbs en las discontinuidades.

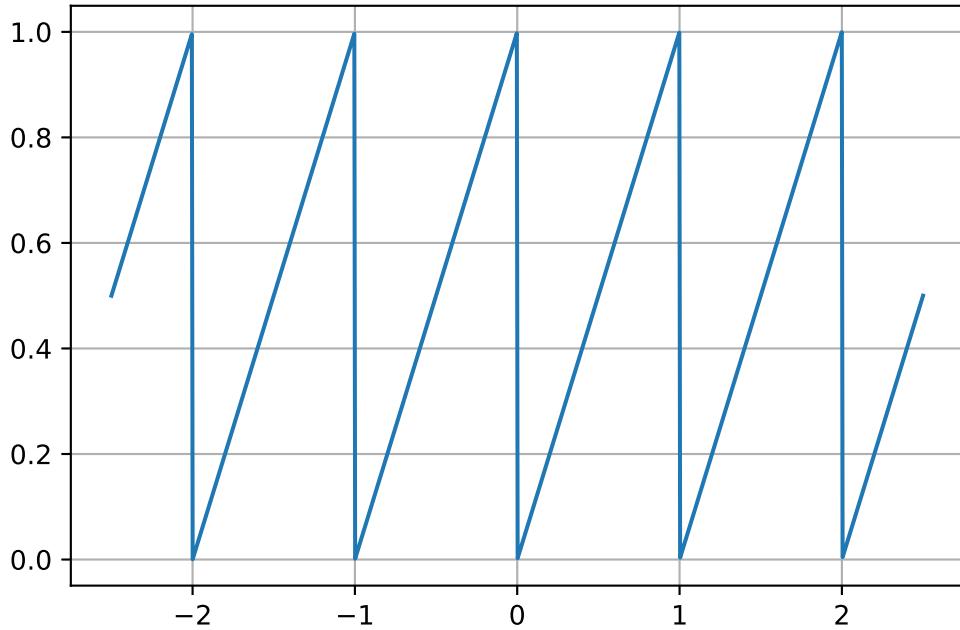


Figura 1.5.: Función sierra.

1.6. ¿Convergencia? Sí, con las condiciones de Dirichlet

Siempre que tratamos con series, al ser sumas infinitas, hay que tener siempre cuidado con la convergencia de la serie, no solo que no diverja, sino a la función que queremos expresar. Las condiciones para que esto se dé son conocidas condiciones de Dirichlet:

1. La función $f(t)$ tiene un número finito de discontinuidades en un período.
2. La función $f(t)$ tiene un número finito de máximos y mínimos en un período.
3. La integral del valor absoluto de $f(t)$ en un período es finita:

$$\int_{-T/2}^{T/2} |f(t)| dt < \infty.$$

¿Qué sucede con las discontinuidades? Las discontinuidades no causan problema como se vio en el ejemplo. En la discontinuidad, la serie converge al promedio de antes y después de

1. Series de Fourier

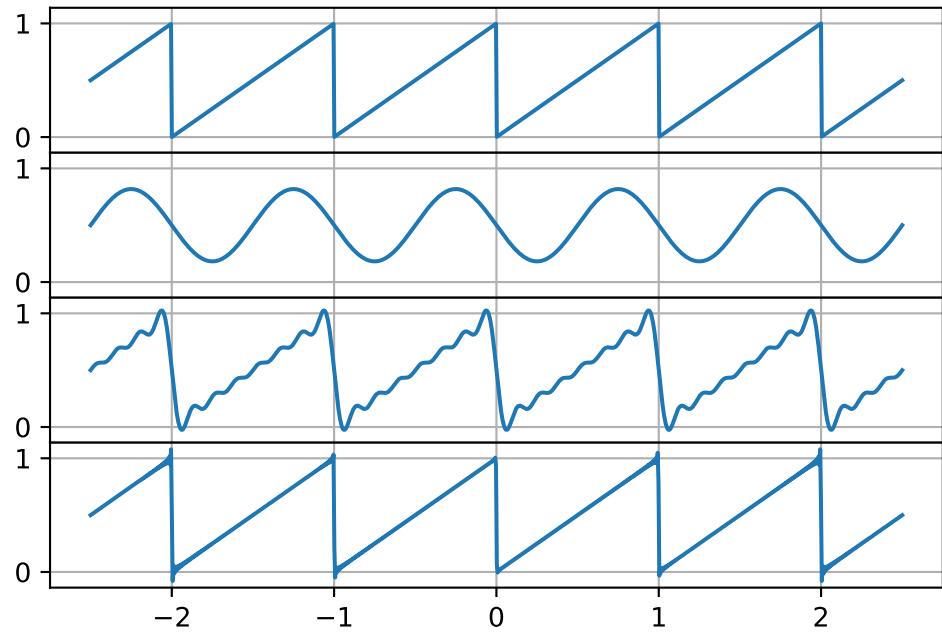


Figura 1.6.: Sumas parciales comparadas con la función original. La primera gráfica muestra la función original, la siguiente la suma parcial con tan solo 2 términos, la tercera con 8 y la última con 100.

las discontinuidades. Si la discontinuidad se da en t_0 :

$$\frac{1}{2} (f(t_0^-) + f(t_0^+))$$

donde $f(t_0^-)$ y $f(t_0^+)$ son los límites izquierdos y derecho de f en t_0 .

1.7. ¿Cómo las podemos utilizar?

Las aplicaciones de las series de Fourier son muy amplias. La primera es de la termodinámica en la difusión de calor. Veamos de qué se trata.

1.7.1. Ecuación de calor

La primera vez que Joseph Fourier utilizó series de funciones en funciones trigonométricas fue para estudiar cómo varía la temperatura en determinada región, en los años 1820's.

Para esto, repasemos un poco de vocabulario. Vamos a considerar que el calor es una transferencia de energía debida a la diferencia de temperatura. Dado que es una transferencia, es usual hablar de flujo de calor y será la cantidad que nos interese. Ahora, la temperatura es proporcional a la energía cinética de los átomos en la sustancia. Una sustancia en el cero absoluto no puede transferir energía. Así pues, Fourier se basó en el principio enunciado por Newton que dicta:

Una diferencia de temperatura entre dos sustancias en contacto, la una con la otra, provoca una transferencia de energía desde la sustancia de mayor temperatura hacia la sustancia de menor temperatura, y eso es el calor o el flujo de calor. Si no hay diferencia de temperatura, no hay calor.

Así pues, vamos a considerar a la temperatura u como una función que depende del espacio x y del tiempo t , $u(x, t)$. Entonces podemos escribir en lenguaje matemático el principio de Newton como una ecuación diferencial en derivadas parciales. Para hacer esto, introducimos el calor que fluye por segundo en x y t como $q(x, t)$ y así tenemos la ley de enfriamiento de Newton

$$q(x, t) = -k \frac{\partial u}{\partial x}, \text{ con } k > 0.$$

La razón del signo menos es que la temperatura siempre es positiva, entonces si la temperatura aumenta, entonces la razón a la que fluye el calor es negativa. La constante k es conocida como resistencia térmica y depende del material. Par un gradiente de temperatura dado, a mayor resistencia, menor es el flujo de calor por segundo, y similarmente, a menor resistencia, mayor es el flujo de calor por segundo.

Mientras el calor fluye de la parte más caliente a la más fría, la temperatura aumenta en la parte más fría de la sustancia. La razón a la que la temperatura crece en un punto x dado por $\frac{\partial u}{\partial t}$ es proporcional a la razón a la que el calor se acumula por unidad de espacio. Ahora, $q(x, t)$ es ya una razón (el flujo de calor por segundo), así pues, podemos escribir que

$$\frac{\partial u}{\partial t} = -k' \frac{\partial q}{\partial x}, \text{ con } k' > 0.$$

La constante k' puede identificarse con la capacidad térmica por unidad espacial. Estas constantes pueden ser asociadas al calor específico. Así, ahora obtenemos la ecuación de calor en una dimensión espacial

$$\frac{\partial u}{\partial t} = kk' \frac{\partial^2 u}{\partial x^2}.$$

Esta ecuación, como se estudia en un curso de ecuaciones diferenciales parciales, tiene que contemplar

- una región espacial

1. Series de Fourier

- una distribución inicial de la temperatura.

Vamos entonces a tratar de resolver esta ecuación en un anillo.

1.7.2. Solución de la ecuación de calor en un anillo por medio de series de Fourier

Vamos a suponer que un anillo es calentado, no necesariamente de forma uniforme, lo que nos da una distribución de temperatura inicial. Esto induce a un flujo de calor alrededor del anillo, lo que provoca un cambio de temperatura en el tiempo. En cualquier tiempo determinado, la temperatura debe ser una función periódica de la posición del círculo.

Podemos manipular las unidades para que el anillo tenga longitud 1 y así $0 \leq x \leq 1$ y también lo haremos tal que la ecuación de calor sea

$$\frac{\partial u}{\partial t} = \frac{1}{2} \frac{\partial^2 u}{\partial x^2}. \quad (1.9)$$

La función u es entonces periódica espacialmente, con período 1 (lo que hace que $\omega_0 = 2\pi/T = 2\pi$), es decir, $u(x+1, t) = u(x, t)$ y podemos expandirla en una serie de Fourier donde los coeficientes dependen del tiempo

$$u(x, t) = \sum_{n=-\infty}^{\infty} c_n(t) e^{2\pi i n x} \text{ donde } c_n(t) = \int_0^1 e^{-2\pi i n x} u(x, t) dx.$$

Así pues, sustituimos en la ecuación 1.9 y tenemos que

$$\begin{aligned} \frac{\partial u}{\partial t} &= \sum_{n=-\infty}^{\infty} c'_n(t) e^{2\pi i n x} \\ &= \sum_{n=-\infty}^{\infty} \left(\int_0^1 e^{-2\pi i n x} \frac{\partial u}{\partial t} dx \right) e^{2\pi i n x} \\ &= \sum_{n=-\infty}^{\infty} \left(\frac{1}{2} \int_0^1 e^{-2\pi i n x} \frac{\partial^2 u}{\partial x^2} dx \right) e^{2\pi i n x} \\ &= \sum_{n=-\infty}^{\infty} \left(-\frac{1}{2} \int_0^1 \frac{d}{dx} (e^{-2\pi i n x}) \frac{\partial u}{\partial x} dx \right) e^{2\pi i n x} \\ &= \sum_{n=-\infty}^{\infty} \left(\frac{1}{2} \int_0^1 \frac{d^2}{dx^2} (e^{-2\pi i n x}) u(x, t) dx \right) e^{2\pi i n x} \\ &= \sum_{n=-\infty}^{\infty} \left(\frac{1}{2} \int_0^1 \frac{d^2}{dx^2} (e^{-2\pi i n x}) u(x, t) dx \right) e^{2\pi i n x} \\ &= \sum_{n=-\infty}^{\infty} \left(\frac{1}{2} \int_0^1 (-4\pi^2 n^2) e^{-2\pi i n x} u(x, t) dx \right) e^{2\pi i n x} \\ &= \sum_{n=-\infty}^{\infty} -2\pi^2 n^2 \left(\int_0^1 u(x, t) e^{-2\pi i n x} dx \right) e^{2\pi i n x} \\ \frac{\partial u}{\partial t} &= \sum_{n=-\infty}^{\infty} -2\pi^2 n^2 c_n(t) e^{2\pi i n x}. \end{aligned}$$

Así obtenemos la relación

$$c'_n(t) = -2\pi^2 n^2 c_n(t),$$

que es una ecuación diferencial ordinaria, cuya solución es

$$c_n(t) = c_n(0) e^{-2\pi^2 n^2 t}.$$

Los valores de $c_n(0)$ se obtienen de la distribución de calor inicial, que podemos decir que es tal que

$$u(x, 0) = f(x) \text{ con } f(x+1) = f(x).$$

Así, utilizando la representación integral de $c_n(t)$, tenemos que

$$\begin{aligned} c_n(0) &= \int_0^1 u(x, 0) e^{-2\pi i n x} dx \\ &= \int_0^1 f(x) e^{-2\pi i n x} dx \\ c_n(0) &= \hat{f}(n), \end{aligned}$$

y así, entonces

$$c_n(t) = \hat{f}(n) e^{-2\pi^2 n^2 t},$$

y entonces la solución es

$$u(x, t) = \sum_{n=-\infty}^{\infty} \hat{f}(n) e^{-2\pi^2 n^2 t} e^{2\pi i n x}.$$

Esta es la solución general de la ecuación de calor.

1.8. Períodos, frecuencias y espectro

Vamos a discutir ahora algunos aspectos sobre los períodos y las frecuencias de las funciones periódicas y así introducimos el término de espectro. Empezamos centrándonos en el concepto geométrico de período y poco a poco fuimos introduciendo el concepto de frecuencia. Resulta natural utilizar el concepto de período para la representación en series de Fourier de $f(t)$,

$$f(t) = \sum_{n=-\infty}^{\infty} \hat{f}(n) e^{2\pi i n t}.$$

El período aquí es 1. Así los términos individuales, para $n \neq 0$, tienen períodos menores a 1:

$$\omega_0 = 2\pi n = \frac{2\pi}{T},$$

por lo que

$$T = \frac{1}{n}.$$

1. Series de Fourier

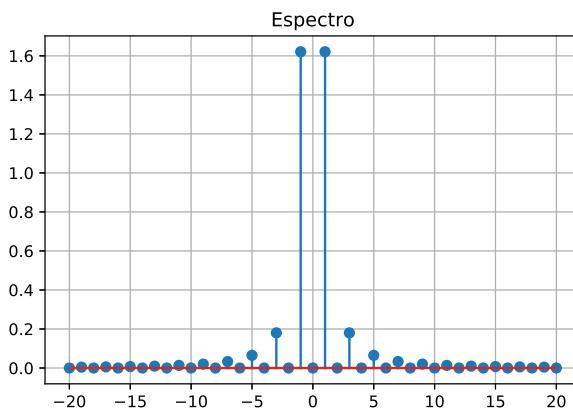


Figura 1.7.: Espectro del primer ejemplo.

Sin embargo, hablar de frecuencias parece ser lo más natural, sin embargo, estamos perdiendo de vista todo el objetivo de realizar este trabajo. Las series de Fourier descomponen a funciones en sus modos de vibración que están caracterizados por sus frecuencias. Las frecuencias presentes en una señal periódica componen el espectro de la señal, de forma más sencilla, los valores n que son los que definen a los coeficientes $\hat{f}(n)$.

Dado que $\hat{f}(-n) = (\hat{f}(n))^*$, los coeficientes de Fourier son o bien $\hat{f}(n) = \hat{f}(-n) = 0$ o ambos son no nulos. Existen señales para las cuales, para $|n| > N$, $\hat{f}(n) = 0$. Para estas señales, decimos que el ancho de banda es N (o $2N$, dependiendo de quién lo diga).

Un espectro se visualiza como vemos en la figura 1.7. Aquí, se colocan en el eje de las abscisas el valor de n y en el eje de las ordenadas el valor de $|c_n|^2$. Este es un espectro de energía. Es muy útil a la hora de comparar dos señales.

1.9. Generalización a más dimensiones

Es posible generalizar a las series de Fourier a 2 y más dimensiones. Aquí, debemos cambiar un poco la interpretación física de las variables: un tiempo multidimensional no tiene mucho sentido, hasta donde sabemos. Estas generalizaciones sirven para periodicidad espacial. Aquí la variable original es $\vec{x} \in \mathbb{R}^n$ y la variable de “frecuencia” debe ser también un vector, que usualmente se denota por $\vec{k} \in \mathbb{R}^n$. Entonces, esto se realiza de forma bastante sencilla de la siguiente manera, la ecuación 1.5 se vuelve entonces la serie de Fourier n -dimensional

$$f(\vec{x}) = \sum_{\vec{k}} c_{\vec{k}} e^{2\pi i \vec{k} \cdot \vec{x}},$$

1.9. Generalización a más dimensiones

donde el producto se convierte entonces en un producto escalar. Los coeficientes son entonces

$$c_{\vec{k}} = \frac{1}{V} \int_V f(\vec{x}) e^{2\pi i n \vec{k} \cdot \vec{x}} d^3x,$$

donde V es el volumen de integración.

2. Transformada de Fourier

2.1. Retomando la idea de la serie de Fourier con un ejemplo y tomando el espectro

Haremos ahora la transición entre las series y la transformada de Fourier. La necesidad de utilizar la transformada de Fourier nace de la necesidad de tratar también con funciones que no son periódicas, para esto tomamos el límite en el que la función tiene un período cada vez más largo.

Empecemos nuestra discusión con un ejemplo bastante sencillo, la función rectángulo que está definida por

$$\Pi(t) = \begin{cases} 1 & |t| < 1/2 \\ 0 & |t| \geq 1/2. \end{cases}$$

Veamos la gráfica en la figura 2.1. Como puede verse no es una función muy complicada. Es una función par y nos sirve para modelar una señal en la que el interruptor se enciende por un segundo y se vuelve a apagar. La función $\Pi(t)$ también se conoce como función sombrero por su gráfica.

Claramente, esta función no es periódica. Por lo tanto, no tiene sentido decir que tiene una serie de Fourier. Así pues, realizaremos un proceso conocido como periodización. Así que repetiremos con intervalos regulares los valores que son distintos de cero. Veamos la gráfica si tomamos como período de $\Pi(t)$, 15, en la figura 2.2.

Ahora que ya tenemos una función periódica, podemos calcular entonces su serie de Fourier. Vamos a dejar libre el período, para fines que veremos más adelante.

La serie de Fourier para una función de período T es

$$f(t) = \sum_{n=1}^{\infty} \hat{f}(n) e^{i\omega_0 nt}, \text{ con } \omega_0 = \frac{2\pi}{T}.$$

Entonces, tenemos que, si $n \neq 0$,

$$\begin{aligned} \hat{f}(n) &= \frac{1}{T} \int_{-T/2}^{T/2} f(t) e^{in\omega_0 t} dt \\ &= \frac{1}{T} \int_{-1/2}^{1/2} e^{in\omega_0 t} dt \\ &= \frac{1}{T} \left[\frac{e^{in\omega_0 t}}{in\omega_0} \right]_{-1/2}^{1/2} \\ \hat{f}(n) &= \frac{1}{T} \left[\frac{e^{\frac{i\omega_0}{2}} - e^{-\frac{i\omega_0}{2}}}{in\omega_0} \right] = \frac{1}{T} \frac{2}{n\omega_0} \sin\left(\frac{n\omega_0}{2}\right) = \frac{1}{n\pi} \sin\left(\frac{n\pi}{T}\right). \end{aligned}$$

2. Transformada de Fourier

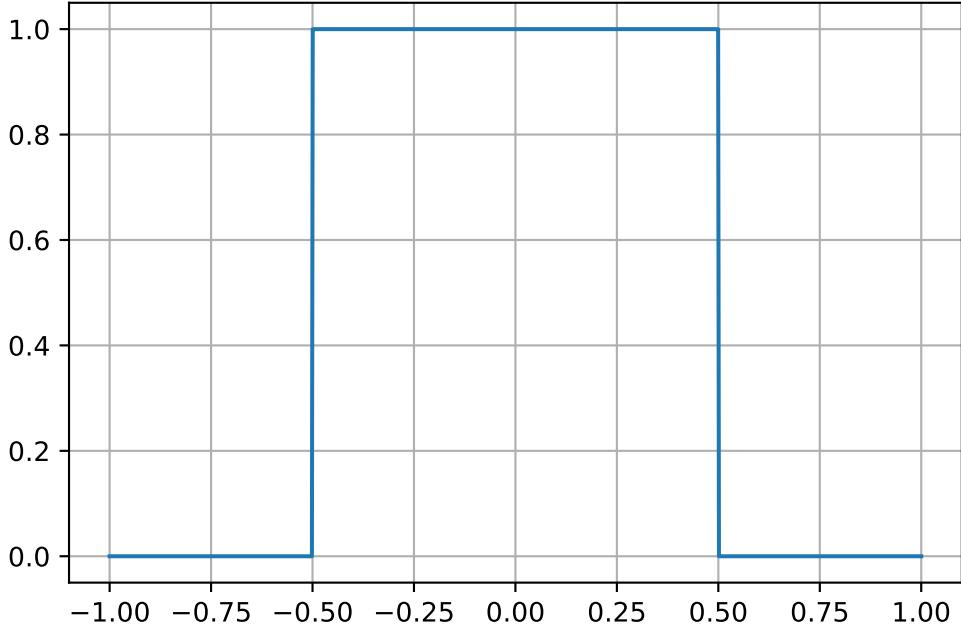


Figura 2.1.: Función rectángulo.

Ahora bien, si $n = 0$,

$$\hat{f}(0) = 1.$$

Veamos ahora los coeficientes de Fourier de la función rectángulo periodizada para distintos períodos, en la figura 2.3.

Vemos que con el aumento del período, las frecuencias se acercan cada vez más y cada vez nos acercamos más a una curva como tal. Vamos a tratar de analizar qué sucede si el período crece ahora indefinidamente.

Recordemos ahora que una serie de Fourier es de la forma

$$f(t) = \sum_{n=-\infty}^{\infty} c_n e^{i\omega_0 n t} = \sum_{n=-\infty}^{\infty} c_n e^{\frac{2\pi i n}{T} t},$$

tal que las frecuencias son entonces $0, \pm \frac{1}{T}, \pm \frac{2}{T}, \dots$. Los puntos en el espectro están separados por $\frac{1}{T}$ y esto provoca que la densidad de puntos sea mayor cuando T aumenta. Recordemos que los coeficientes de Fourier están entonces dados por $\frac{1}{n\pi} \sin\left(\frac{n\pi}{T}\right)$, pero normalizamos en la figura 2.3, utilizando como coeficientes $\frac{T}{n\pi} \sin\left(\frac{n\pi}{T}\right)$.

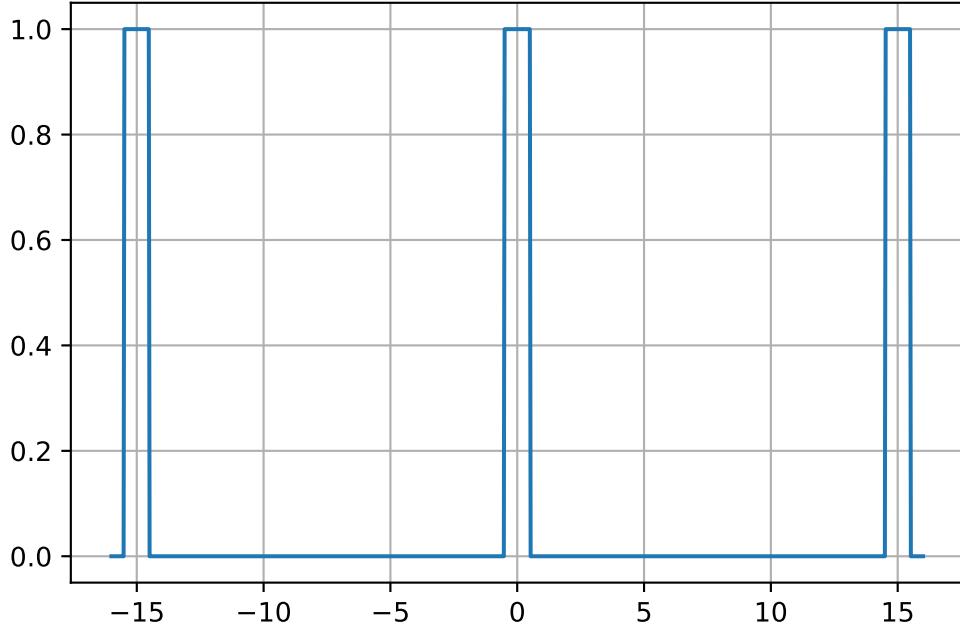


Figura 2.2.: Periodización de la función rectángulo, con período 15.

2.2. La transformada de Fourier

Nuestro siguiente paso es tomar a $\frac{n}{T}$ que es una variable discreta, de la cual dependen nuestros coeficientes de Fourier y mientras T crece, podemos tomar el límite hasta que tengamos una variable continua $s = \frac{n}{T}$ para tener así que los coeficientes son de la forma

$$\frac{\sin(\pi s)}{\pi s},$$

y así, si tomamos el límite cuando $T \rightarrow \infty$, tenemos que la serie de Fourier se vuelve en una integral de la forma

$$\hat{\Pi}(s) = \int_{-\infty}^{\infty} e^{-2\pi i st} \Pi(t) dt.$$

Esta es la transformada de Fourier. Para nuestro ejemplo,

$$\hat{\Pi}(s) = \int_{-\infty}^{\infty} e^{-2\pi i st} \Pi(t) dt = \int_{-1/2}^{1/2} e^{-2\pi i st} dt = \frac{\sin(\pi s)}{\pi s},$$

2. Transformada de Fourier

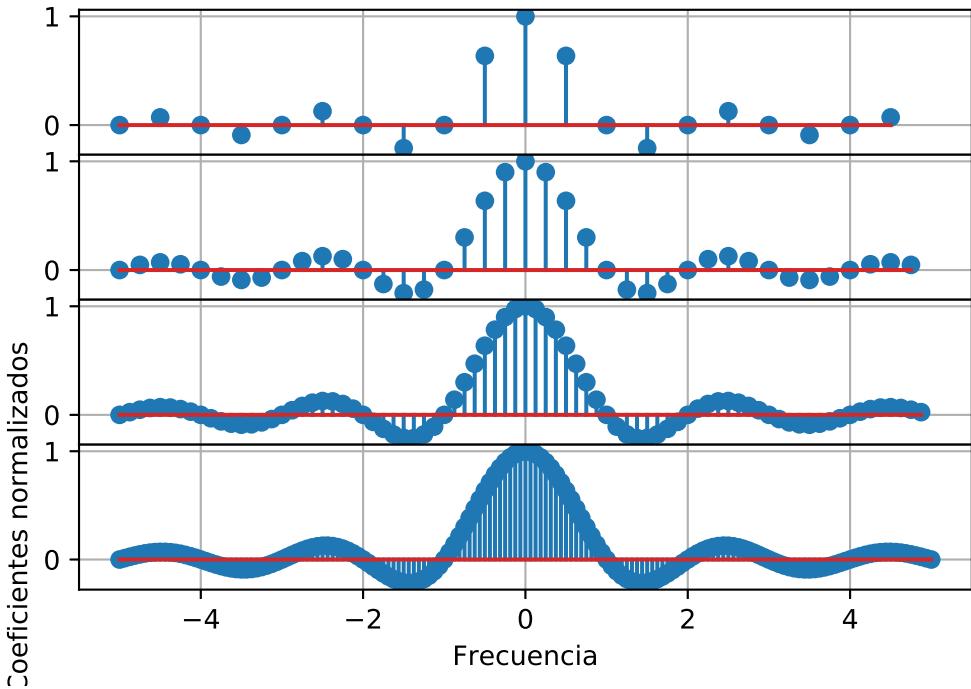


Figura 2.3.: Función rectángulo periodizada. La de arriba con período 2, la siguiente con período 4, luego con período 8 y por último con período 16.

conocida como función sinc, cuya gráfica mostramos en la figura 2.4.

Así entonces, para una función f cualquiera, la transformada de Fourier es

$$\hat{f}(s) = \int_{-\infty}^{\infty} e^{-2\pi i s t} f(t) dt.$$

Ahora ya se puede ver el por qué de la notación $\hat{f}(n)$ para los coeficientes de Fourier de las ecuaciones 1.6 y 1.7.

2.3. Algunos comentarios sobre la transformada de Fourier

La primera pregunta que debemos hacernos es sobre la convergencia de la transformada. Vamos a ocuparnos de este asunto luego, así que asumiremos, por el momento que la integral converge y por lo tanto existe.

2.3. Algunos comentarios sobre la transformada de Fourier

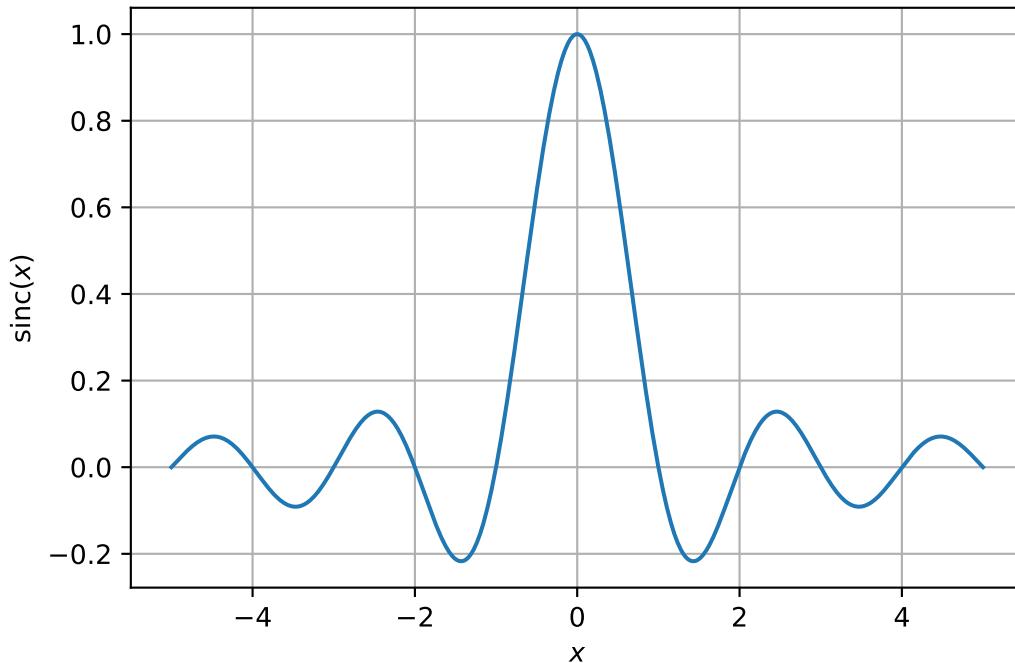


Figura 2.4.: Función sinc.

Una función exponencial, físicamente, debe ser adimensional. Por lo tanto, s debe tener como dimensionales 1/tiempo.

Mientras la transformada de Fourier toma importancia al querer tener información espectral de una función no-periódica. ¿Cómo es esto posible? Tratemos de profundizar un poco en el asunto. Las series de Fourier poseen un espectro con un conjunto de frecuencias discretas, contrario a la transformada de Fourier que tiene un espectro continuo, i.e. un continuo de frecuencias.

Así vemos entonces que la transformada de Fourier analiza a una señal en sus componentes de frecuencia, en lugar de tiempo. Resulta natural que si vamos a analizar o manipular a nuestra señal en esta base, ¿cómo podemos hacer para regresar a la señal original? Existe una transformada inversa de Fourier.

2. Transformada de Fourier

2.4. Transformada inversa de Fourier

Buscamos entonces ahora obtener a $f(t)$ a partir de $\hat{f}(s)$. Para entender la idea central de todo esto, vamos a suponer que $f(t)$ se anula afuera de cierto intervalo (al igual que la función rectángulo de la figura 2.1) y vamos a considerar que tenemos un período T que es grande. Entonces expandimos $f(t)$ en su serie de Fourier,

$$f(t) = \sum_{n=-\infty}^{\infty} c_n e^{\frac{2\pi i n t}{T}},$$

y los coeficientes de Fourier pueden escribirse a través de la transformada de Fourier de f en los puntos $s_n = \frac{n}{T}$.

$$c_n = \frac{1}{T} \int_{-T/2}^{T/2} e^{-\frac{2\pi i n t}{T}} f(t) dt = \frac{1}{T} \int_{-\infty}^{\infty} e^{-\frac{2\pi i n t}{T}} f(t) dt.$$

La segunda igualdad es porque por hipótesis, $f(t) = 0$ fuera del intervalo $[-T/2, T/2]$, por lo que

$$c_n = \frac{1}{T} \hat{f}\left(\frac{n}{T}\right) = \frac{1}{T} \hat{f}(s_n).$$

Si insertamos esto ahora en la expresión de $f(t)$, obtenemos

$$f(t) = \sum_{n=-\infty}^{\infty} \frac{1}{T} \hat{f}(s_n) e^{2\pi i s_n t}.$$

Ahora consideramos que los puntos $s_n = \frac{n}{T}$ están separados por $\frac{1}{T}$, así que podemos definir a $\Delta s \equiv \frac{1}{T}$, y así podemos aproximar por una integral

$$\sum_{n=-\infty}^{\infty} \frac{1}{T} \hat{f}(s_n) e^{2\pi i s_n t} = \sum_{n=-\infty}^{\infty} \hat{f}(s_n) e^{2\pi i s_n t} \Delta s \approx \int_{-\infty}^{\infty} \hat{f}(s) e^{2\pi i s t} ds.$$

Hemos encontrado, entonces, la transformada inversa de Fourier

$$f(t) = \int_{-\infty}^{\infty} \hat{f}(s) e^{2\pi i s t} ds.$$

Podemos entonces recuperar la expresión de $f(t)$ a partir de $\hat{f}(s)$. Para una función $g(s)$, vamos a denotar a su transformada inversa de Fourier como

$$\check{g}(t) = \int_{-\infty}^{\infty} e^{2\pi i s t} g(s) ds.$$

De nuevo, asumiremos que la integral existe y nos preocuparemos por su convergencia más adelante.

Veamos ahora un teorema importante.

Teorema 2.1. *Teorema de inversión de Fourier.*

$$f(t) = \int_{-\infty}^{\infty} e^{2\pi i st} \hat{f}(s) ds,$$

o escrito de otra manera,

$$\check{\hat{f}} = f.$$

En palabras, la transformada inversa de Fourier de la transformada de Fourier es la función original.

Inversamente, también tenemos que

$$\hat{\check{g}} = g.$$

Notemos que la transformada inversa de Fourier se ve exactamente igual que la transformada de Fourier, con la excepción de un signo menos. Vamos a discutir esta singularidad más adelante.

También es importante recalcar otra propiedad importante, acerca del espectro de potencia $|\hat{f}(s)|^2$ o el espectro de densidad de potencia o espectro de energía, que se conoce como la identidad de Parseval para transformadas de Fourier.

$$\int_{-\infty}^{\infty} |f(t)|^2 dt = \int_{-\infty}^{\infty} |\hat{f}(s)|^2 ds.$$

Con esto, inferimos que requerimos entonces que la integral $\int_{-\infty}^{\infty} f(t) dt$ esté definida para que la transformada de Fourier y su inversa existan.

Notaciones Vamos a cambiar un poco las notaciones ya que las notaciones de $\check{\cdot}$ y $\hat{\cdot}$ pueden llegar a ser confusas. Vamos a denotar a la transformada de Fourier de la función f como

$$(\mathcal{F}f)(s) = \int_{-\infty}^{\infty} e^{-2\pi i st} f(t) dt.$$

Así también, la transformada inversa de Fourier de la función g se denota como

$$(\mathcal{F}^{-1}g)(t) = \int_{-\infty}^{\infty} e^{2\pi i st} g(s) ds.$$

2.5. Propiedades generales

Veamos algunas propiedades generales de las transformadas de Fourier.

2.5.1. Dualidades

La transformada \mathcal{F} y la transformada inversa \mathcal{F}^{-1} de Fourier son muy similares, excepto por un signo en la exponencial. Esto tiene como consecuencia que podemos sustituir a s

2. Transformada de Fourier

por $-s$ en la definición de la transformada de Fourier y tenemos la transformada inversa y viceversa, veamos esto en lenguaje matemático

$$\begin{aligned} (\mathcal{F}f)(-s) &= \int_{-\infty}^{\infty} e^{-2\pi i(-s)t} f(t) dt = \int_{-\infty}^{\infty} e^{2\pi ist} f(t) dt = (\mathcal{F}^{-1}f)(s) \\ (\mathcal{F}^{-1}f)(-t) &= \int_{-\infty}^{\infty} e^{-2\pi is(-t)} f(s) ds = \int_{-\infty}^{\infty} e^{-2\pi ist} f(s) ds = (\mathcal{F}f)(t) \end{aligned} \quad (2.1)$$

Esta propiedad nos facilitará mucho los cálculos más adelante. Veamos una aplicación de esto calculando la transformada de Fourier de la función sinc,

$$\mathcal{F}\Pi = \text{sinc}$$

y por lo tanto

$$\mathcal{F}^{-1}\text{sinc} = \Pi$$

y por lo tanto

$$(\mathcal{F}\text{sinc})(t) = (\mathcal{F}^{-1}\text{sinc})(-t) = \Pi(-t) = \Pi(t).$$

Hemos probado entonces que $\mathcal{F}\text{sinc} = \Pi$.

De estas propiedades, podemos ver que solo *ponerle un signo menos* a la variable, permite un paso muy sencillo entre transformaciones de Fourier y las transformaciones inversas de Fourier.

Definición 2.1. Si tenemos una señal descrita por una función $f(t)$ entonces la señal invertida, está descrita por la función f^- definida por

$$f^-(t) = f(-t).$$

Notemos entonces que las señales invertidas tienen un tratamiento muy sencillo con las funciones pares e impares ya que

$$\begin{aligned} f \text{ es par si } f^- &= f \\ f \text{ es impar si } f^- &= -f. \end{aligned}$$

Entonces ahora veamos cómo se aplica esto a las transformadas de Fourier,

$$(\mathcal{F}f)^- = \mathcal{F}^{-1}f$$

y de igual manera

$$(\mathcal{F}^{-1}f)^- = \mathcal{F}f.$$

Estas dos ecuaciones, básicamente nos dicen que la transformada inversa de Fourier no es más que la transformada de Fourier invertida (invertida en el sentido de las señales invertidas).

2.5.2. Invariantes de paridad y de conjugación compleja

Buscamos ahora responder a una pregunta que resulta natural que es: ¿Cómo se reflejan las simetrías de paridad de las señales en las transformadas de Fourier?

Proposición 2.1. *Si $f(x)$ es par o impar, entonces la transformada de Fourier es par o impar, respectivamente:*

$$\begin{aligned} \text{si } f \text{ es par } (\mathcal{F}f)^- &= \mathcal{F}f \\ \text{si } f \text{ es impar } (\mathcal{F}f)^- &= -\mathcal{F}f, \end{aligned}$$

y se deduce entonces que

$$(\mathcal{F}f)^- = \mathcal{F}f^- = \begin{cases} \mathcal{F}f & \text{si } f \text{ es par} \\ \mathcal{F}(-f) = -\mathcal{F}f & \text{si } f \text{ es impar.} \end{cases}$$

Ahora bien, recordemos que la transformada de Fourier es una función de variable compleja, así que debemos considerar qué sucede con la conjugación compleja.

Proposición 2.2. *Si $f(t)$ es una función real, entonces $(\mathcal{F}f)^- = \overline{\mathcal{F}f}$ y también $\mathcal{F}(f^-) = \overline{\mathcal{F}f}$.*

Las pruebas de estas proposiciones son sencillas a partir de las definiciones y se dejan como ejercicio para el lector. Pasemos a otras propiedades sencillas que combinan a estas dos propiedades.

Proposición 2.3. *Si f es real y par, entonces su transformada de Fourier es real.*

Si f es real e impar, entonces su transformada de Fourier es impar e imaginaria pura.

Demostración. Tomemos primero el caso en el que f sea una función par, entonces debemos probar que $\overline{\mathcal{F}f} = \mathcal{F}f$. Entonces calculemos

$$\overline{\mathcal{F}f} = (\mathcal{F}f)^- = \mathcal{F}f.$$

Ahora tomemos el caso en que f es impar, entonces probemos que $\overline{\mathcal{F}f} = -\mathcal{F}f$. Entonces calculemos

$$\overline{\mathcal{F}f} = (\mathcal{F}f)^- = -\mathcal{F}f.$$

□

2.5.3. Linealidad

Una propiedad de suma importancia es la de la linealidad: la transformada de Fourier es una transformación lineal, y por lo tanto, la transformada inversa de Fourier, también lo es.

Proposición 2.4. *La transformada de Fourier es una transformación lineal, es decir que para funciones f y g y para un escalar α , se tiene que*

$$\begin{aligned} (\mathcal{F}(f+g))(s) &= (\mathcal{F}f)(s) + (\mathcal{F}g)(s) \\ (\mathcal{F}(\alpha f))(s) &= \alpha(\mathcal{F}f)(s). \end{aligned}$$

2. Transformada de Fourier

Esto es consecuencia inmediata de la definición por integrales de la transformada de Fourier, ya que la integral definida es una transformación lineal.

2.5.4. Teorema de traslación

Una traslación en la variable t tiene una implicación muy sencilla en la transformada de Fourier: una rotación en el plano complejo. Esto quiere decir que la magnitud se mantiene. Antes de establecer el teorema, veamos qué es lo que sucede al calcular la transformada de Fourier de $f(t + b)$, la función f con la variable trasladada por un factor b , entonces, vamos a sustituir la variable t por $u = t + b$,

$$\begin{aligned}\int_{-\infty}^{\infty} f(t+b) e^{-2\pi i st} dt &= \int_{-\infty}^{\infty} f(u) e^{-2\pi i s(u-b)} du \\ &= \int_{-\infty}^{\infty} f(u) e^{-2\pi i su} e^{2\pi i sb} du \\ &= e^{2\pi i sb} \int_{-\infty}^{\infty} f(u) e^{-2\pi i su} du \\ \int_{-\infty}^{\infty} f(t+b) e^{-2\pi i st} dt &= e^{2\pi i sb} (\mathcal{F}f)(s).\end{aligned}$$

Entonces enunciamos el teorema de traslación para transformadas de Fourier.

Proposición 2.5. *Una traslación por un factor b en la variable de una función f se ve reflejada en la función $g(t) = f(t + b)$ como*

$$\mathcal{F}g = e^{2\pi i sb} \mathcal{F}f$$

y entonces

$$|\mathcal{F}f| = |\mathcal{F}g|.$$

2.5.5. Teorema de cambio de escala o de similitud

Aquí la pregunta es: ¿Si realizamos un cambio de escala en la variable cómo cambia la transformada de Fourier? Es decir si conocemos la transformada de Fourier de la función f , ¿qué sucede con la transformada de Fourier de $f(at)$? con a un escalar. Hagamos el cálculo, tomando el cambio de variable de t por $u = at$, si $a > 0$,

$$\begin{aligned}\int_{-\infty}^{\infty} f(at) e^{-2\pi i st} dt &= \int_{-\infty}^{\infty} f(u) e^{-2\pi i s \frac{u}{a}} \frac{1}{a} du \\ &= \frac{1}{a} \int_{-\infty}^{\infty} f(u) e^{-2\pi i su/a} du \\ \int_{-\infty}^{\infty} f(at) e^{-2\pi i st} dt &= \frac{1}{a} (\mathcal{F}f)\left(\frac{s}{a}\right).\end{aligned}$$

Si $a < 0$, los límites de integración se invierten y entonces tenemos que si $a < 0$, $\int_{-\infty}^{\infty} f(at) e^{-2\pi i st} dt = -\frac{1}{a} (\mathcal{F}f)\left(\frac{s}{a}\right)$. Entonces podemos enunciar el teorema de similitud.

Teorema 2.2. *Teorema de similitud. Un cambio de escala por un factor a en la variable de una función f se ve reflejada en la función $g(t) = f(at)$ como*

$$(\mathcal{F}g)(s) = \frac{1}{|a|} (\mathcal{F}f)\left(\frac{s}{a}\right).$$

Notemos entonces que un cambio de escala en la variable t , es decir horizontal, se refleja en un cambio inverso tanto en s como en la transformada, es decir horizontal y verticalmente.

2.5.6. La transformada de Fourier de una derivada

Calculemos la transformada de Fourier de la derivada de f , f' ,

$$(\mathcal{F}f')(s) = \int_{-\infty}^{\infty} f'(t) e^{-2\pi i st} dt = \underbrace{f(t) e^{-2\pi i st}}_{=0} \Big|_{-\infty}^{\infty} - \int_{-\infty}^{\infty} -2\pi i s e^{-2\pi i st} f(t) dt = 2\pi i s (\mathcal{F}f)(s),$$

entonces

$$(\mathcal{F}f')(s) = 2\pi i s (\mathcal{F}f)(s)$$

2.6. Más ejemplos

Vamos a analizar algunos de los ejemplos típicos de señales y analizaremos aquí sus transformadas de Fourier.

2.6.1. Función triángulo

Consideramos la función triángulo que está definida por

$$\Lambda(x) = \begin{cases} 1 - |x| & \text{si } |x| \leq 1 \\ 0 & \text{en cualquier otro caso,} \end{cases}$$

como se ve en la figura 2.1.

Ahora calculemos su transformada de Fourier

$$\begin{aligned} (\mathcal{F}\Lambda)(s) &= \int_{-\infty}^{\infty} \Lambda(x) e^{-2\pi i sx} dx \\ &= \int_{-1}^0 (1+x) e^{-2\pi i sx} dx + \int_0^1 (1-x) e^{-2\pi i sx} dx \\ &= \left(\frac{1+2\pi is}{4\pi^2 s^2} - \frac{e^{2\pi is}}{4\pi^2 s^2} \right) - \left(\frac{2\pi is-1}{4\pi^2 s^2} + \frac{e^{-2\pi is}}{4\pi^2 s^2} \right) \\ &= \frac{e^{-2\pi is}(e^{2\pi is}-1)^2}{4\pi^2 s^2} = \frac{e^{-2\pi is}(e^{\pi is}(e^{\pi is}-e^{-\pi is}))^2}{4\pi^2 s^2} \\ (\mathcal{F}\Lambda)(s) &= \frac{e^{-2\pi is}(e^{\pi is}(2i \sin(\pi s)))^2}{4\pi^2 s^2} = \operatorname{sinc}^2(s). \end{aligned}$$

Este resultado no es sorpresivo. Esto es por la convolución, tema que estudiaremos en secciones siguientes, pero también por la dualidad de la ecuación 2.1, podemos ahora calcular la transformada de sinc^2 . Como consecuencia inmediata al cálculo anterior, tenemos que

$$\mathcal{F}^{-1} \operatorname{sinc}^2 = \Lambda.$$

Entonces

$$(\mathcal{F}\operatorname{sinc}^2)(t) = (\mathcal{F}^{-1}\operatorname{sinc}^2)(-t) = \Lambda(-t) = \Lambda(t).$$

La forma de esta función puede verse en la figura 2.6.

2. Transformada de Fourier

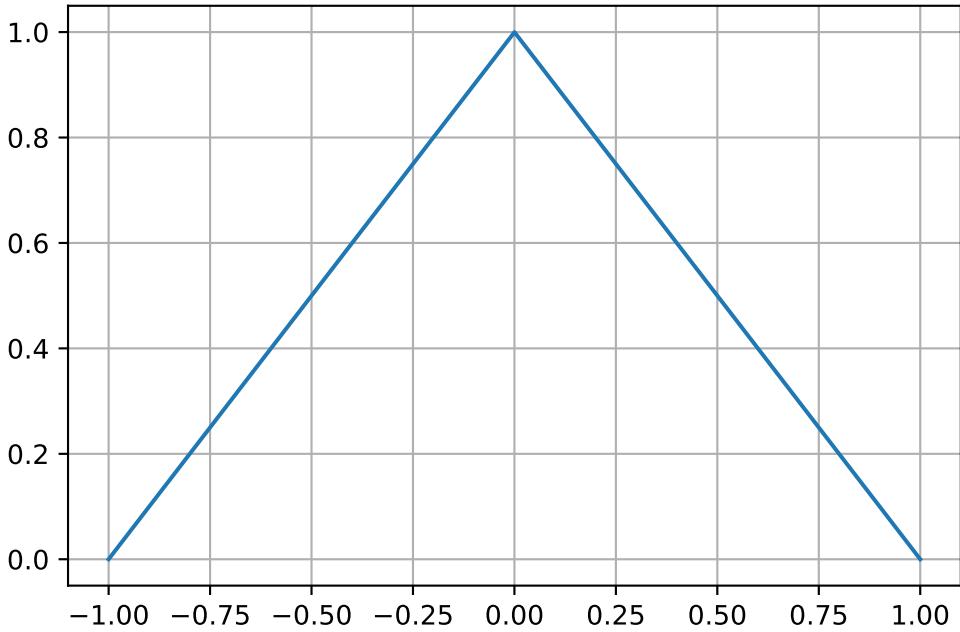


Figura 2.5.: Función triángulo.

2.6.2. Decaimiento exponencial

Otra función que es común en las señales es el decaimiento exponencial, lo definimos por

$$f(t) = \begin{cases} 0 & \text{si } t \leq 0 \\ e^{-at} & \text{si } t > 0, \end{cases}$$

donde a es una constante positiva. Veamos su comportamiento para distintos valores de a en la figura 2.7.

Calculemos ahora su transformada de Fourier

$$\begin{aligned} (\mathcal{F}f)(s) &= \int_{-\infty}^{\infty} e^{-2\pi ist} f(t) dt \\ &= \int_0^{\infty} e^{-2\pi ist} e^{-at} dt \\ &= \int_0^{\infty} e^{-(2\pi is+a)t} dt \\ (\mathcal{F}f)(s) &= \frac{1}{2\pi is+a}. \end{aligned}$$

Aquí, obtuvimos a una función que es compleja, contrario a los ejemplos anteriores. El hecho que para la función rectángulo y triángulo teníamos funciones reales es por el hecho

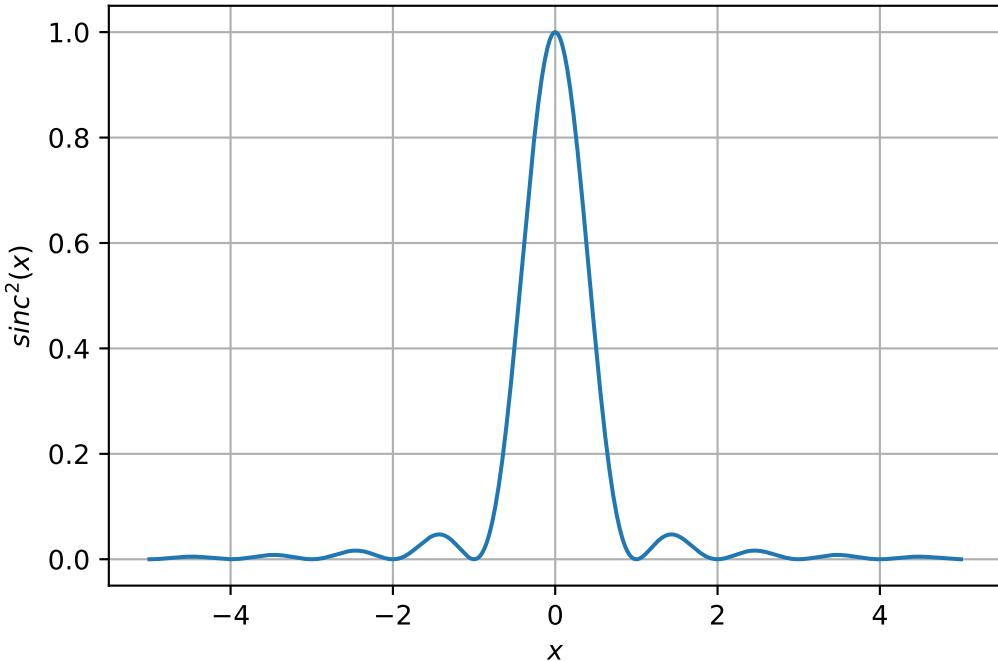


Figura 2.6.: Función $\text{sinc}^2(s)$.

que son funciones pares. Este no es el caso para el decaimiento exponencial. Así pues, para ver el espectro del decaimiento exponencial, vamos a graficar

$$|(\mathcal{F}f)(s)|^2 = \frac{1}{|2\pi is + a|^2} = \frac{1}{a^2 + 4\pi^2 s^2},$$

como se ve en la figura 2.8.

A esta función se le conoce como función lorenziana y viene de analizar la vida media de átomos que son radioactivos.

2.6.3. Distribución gaussiana

Ahora analizaremos otra señal común que es la distribución gaussiana. Una función gaussiana tiene una forma de campana, conocida como la campana de Gauss, que se muestra en la figura 2.9.

Dado el significado estadístico que tiene esta función, es importante que nos detengamos un poco en su estudio y sus propiedades. Como bien se sabe de los cursos de cálculo, esta

2. Transformada de Fourier

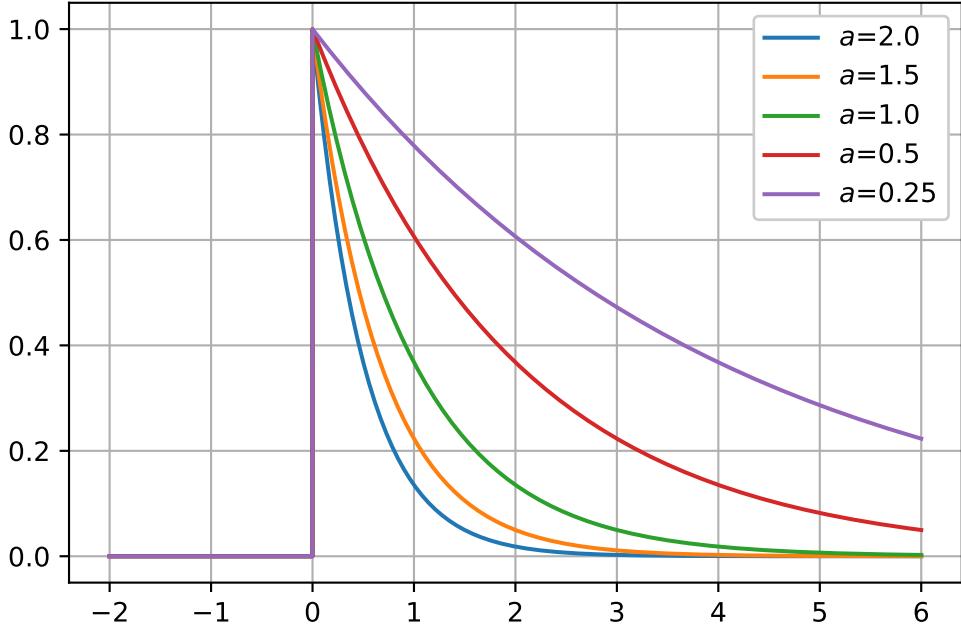


Figura 2.7.: Decaimiento exponencial.

función no tiene una antiderivada definida. Veamos cómo se integra sobre los reales a esta función. Sea I , la integral

$$I = \int_{-\infty}^{\infty} e^{-ax^2} dx.$$

Dado que la integración es sobre la variable x , también podemos escribirla como

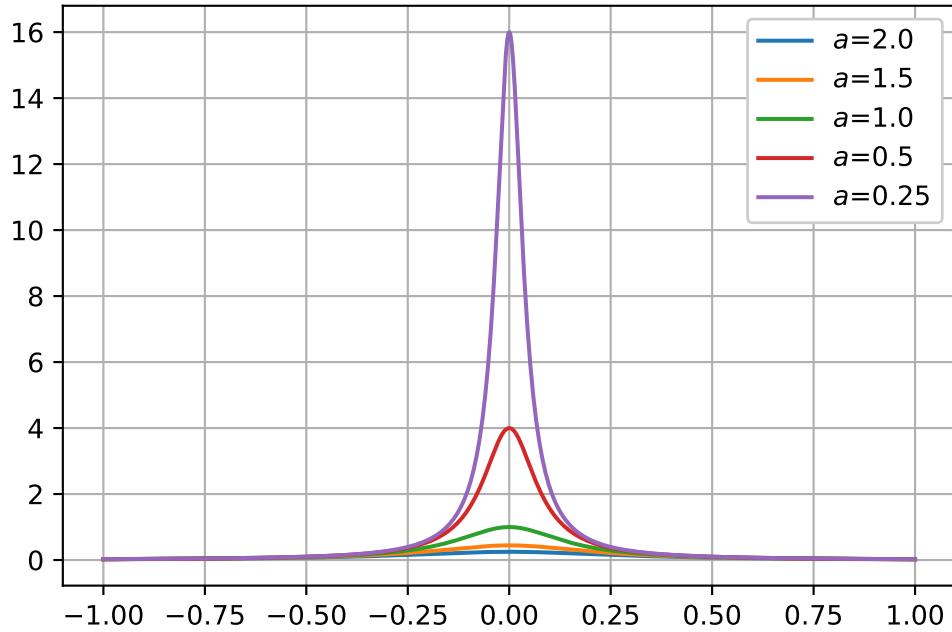
$$I = \int_{-\infty}^{\infty} e^{-ay^2} dy.$$

Entonces, podemos escribir, aunque parezca extraño que

$$I^2 = \left(\int_{-\infty}^{\infty} e^{-ax^2} dx \right) \left(\int_{-\infty}^{\infty} e^{-ay^2} dy \right) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} e^{-a(x^2+y^2)} dx dy.$$

Este es el truco, ahora, podemos utilizar las coordenadas polares

$$r^2 = x^2 + y^2 \text{ y } dx dy = r dr d\theta,$$

**Figura 2.8.:** Función lorenziana.

para tener que la integral es

$$\int_0^{2\pi} \int_0^{\infty} e^{-ar^2} r dr d\theta = 2\pi \int_0^{\infty} r e^{-ar^2} dr = \frac{\pi}{a}$$

Así pues, $I = \int_{-\infty}^{\infty} e^{-x^2} dx = \sqrt{\frac{\pi}{a}}$. Ahora sí, estamos listos para evaluar la transformada de Fourier de la función gaussiana

$$f(x) = e^{-\pi x^2},$$

cuya integral sobre los reales es 1. Entonces la transformada de Fourier es

$$(\mathcal{F}f)(s) = \int_{-\infty}^{\infty} e^{-\pi x^2} e^{-2\pi i s x} dx,$$

que parece bastante complicada de calcular. Para poder hacerlo, notemos que derivada

2. Transformada de Fourier

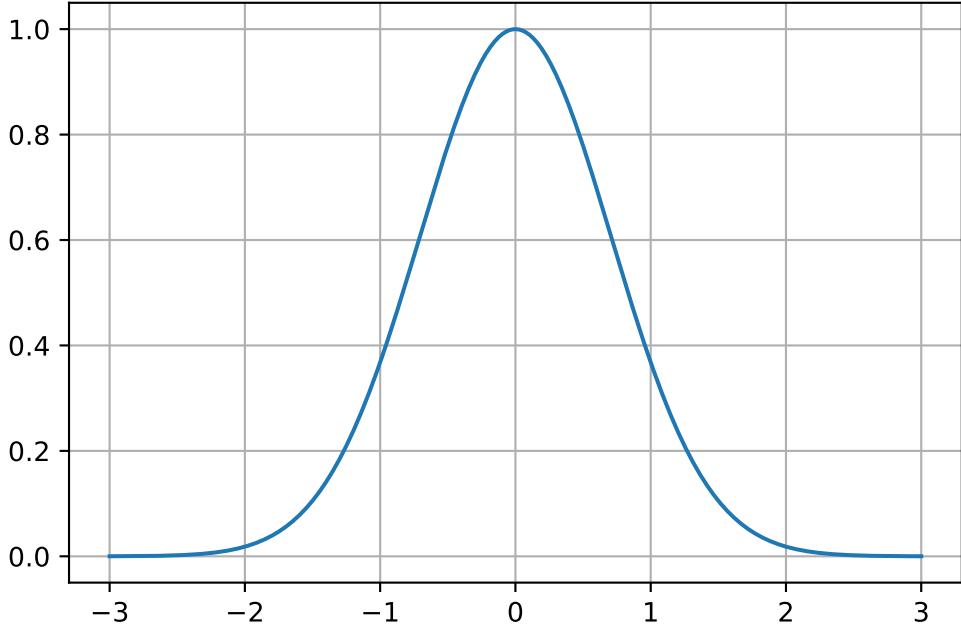


Figura 2.9.: Distribución gaussiana.

respecto a s es

$$\begin{aligned}
 \frac{d}{ds} (\mathcal{F}f)(s) &= \frac{d}{ds} \left(\int_{-\infty}^{\infty} e^{-\pi x^2} e^{-2\pi i s x} dx \right) \\
 &= \int_{-\infty}^{\infty} \frac{d}{ds} \left(e^{-\pi x^2} e^{-2\pi i s x} \right) dx \\
 &= \int_{-\infty}^{\infty} e^{-\pi x^2} (-2\pi i x) e^{-2\pi i s x} dx \\
 &= \underbrace{e^{-2\pi i s x} i e^{-\pi x^2}}_{\stackrel{x \rightarrow -\infty}{=0}} \Big|_{-\infty}^{\infty} - 2\pi s \int_{-\infty}^{\infty} e^{-\pi x^2} e^{-2\pi i s x} dx \\
 \frac{d}{ds} (\mathcal{F}f)(s) &= -2\pi s (\mathcal{F}f)(s).
 \end{aligned}$$

Por lo que $\mathcal{F}f(s)$ satisface la ecuación diferencial

$$\frac{d}{ds} (\mathcal{F}f)(s) = -2\pi s (\mathcal{F}f)(s),$$

con condiciones iniciales

$$(\mathcal{F}f)(s) = (\mathcal{F}f)(0) e^{-\pi s^2},$$

pero

$$(\mathcal{F}f)(0) = \int_{-\infty}^{\infty} e^{-\pi x^2} dx = 1.$$

Finalmente

$$(\mathcal{F}f)(s) = e^{-\pi s^2}.$$

Entonces, la función gaussiana es su propia transformada de Fourier.

2.6.4. Función impulso o delta de Dirac

La función impulso o delta de Dirac no es realmente una función. Es lo que se conoce como una distribución. Su definición es muy particular.

Definición 2.2. Delta de Dirac. La distribución $\delta(x)$ o delta de Dirac es tal que para una función $f(x)$,

$$\int_{-\infty}^{\infty} f(x) \delta(x) dx = f(0). \quad (2.2)$$

Una propiedad inmediata es que

$$\int_{-\infty}^{\infty} \delta(x) dx = 1.$$

Es posible que en la literatura se encuentre una definición alternativa

$$\delta(x) = \begin{cases} \infty & \text{si } x = 0 \\ 0 & \text{en cualquier otro caso.} \end{cases}$$

Esta definición es incorrecta, ya que ∞ no tiene realmente un sentido, pero es útil para darnos una intuición de cómo es la distribución, y se ve en la figura 2.10. Sin embargo, para los cálculos, la definición 2.2.

Calculemos entonces la transformada de Fourier,

$$\begin{aligned} (\mathcal{F}\delta)(s) &= \int_{-\infty}^{\infty} \delta(t) e^{-2\pi i st} dt \\ (\mathcal{F}\delta)(s) &= e^{-2\pi i st} \Big|_{t=0} = 1. \end{aligned}$$

Así, podemos deducir la siguiente propiedad.

Proposición 2.6. Tenemos la siguiente identidad.

$$\delta(t) = \int_{-\infty}^{\infty} e^{2\pi i st} ds.$$

Demostración. Notemos que

$$\begin{aligned} \delta(t) &= (\mathcal{F}^{-1}1)(t) \\ \delta(t) &= \int_{-\infty}^{\infty} e^{2\pi i st} ds. \end{aligned}$$

□

Esta función es muy importante para señales que son discretas y no continuas.

2. Transformada de Fourier

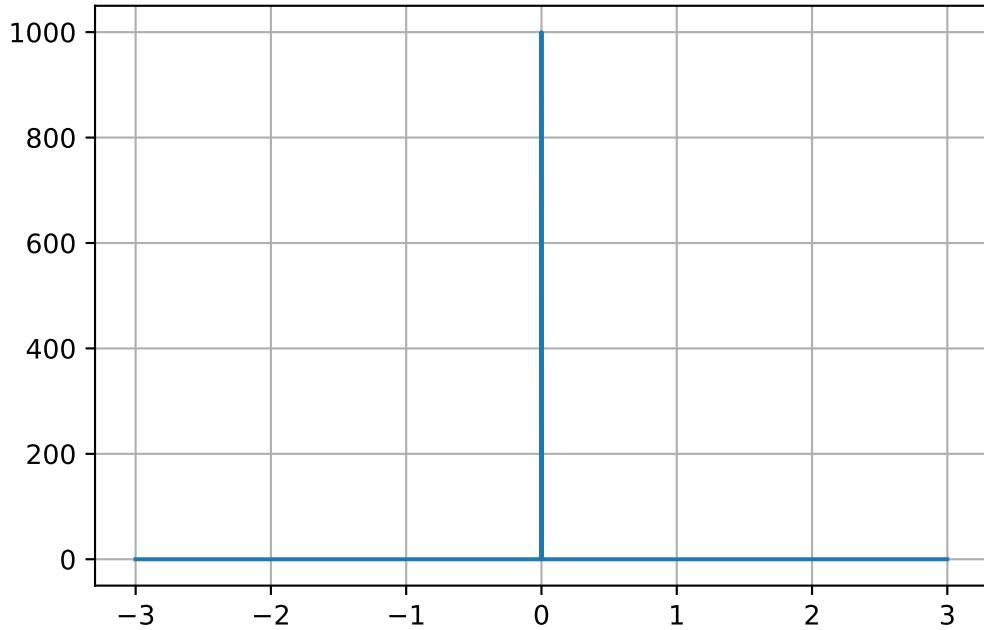


Figura 2.10.: Delta de Dirac: esta es una caricatura de la distribución conocida como delta de Dirac.

2.6.5. Función constante

Ahora vamos a considerar a una función constante $f(t) = c$. Calculemos su transformada de Fourier,

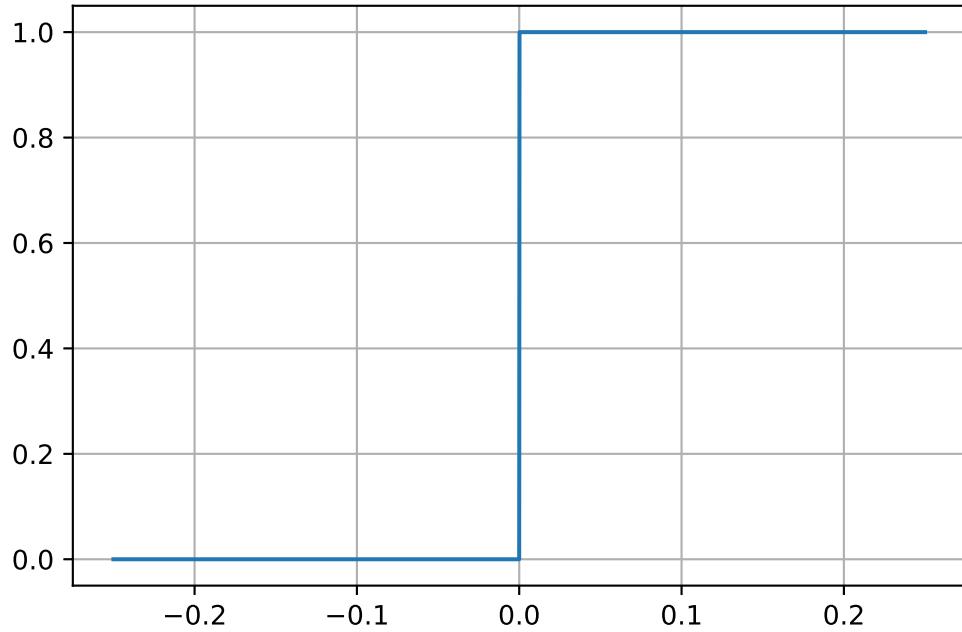
$$\begin{aligned} (\mathcal{F}f)(s) &= \int_{-\infty}^{\infty} f(t) e^{-2\pi i st} dt \\ &= c \int_{-\infty}^{\infty} e^{-2\pi i st} dt \\ (\mathcal{F}f)(s) &= c\delta(-s) = c\delta(s). \end{aligned}$$

Notemos que $\delta(t) = \delta(-t)$, ya que por la definición 2.2, tenemos que

$$\int_{-\infty}^{\infty} f(t) \delta(-t) dt = f(0).$$

Sin embargo, $\delta(t) \neq c\delta(t)$, en general, ya que

$$\int_{-\infty}^{\infty} f(t) c\delta(t) dt = cf(0).$$

**Figura 2.11.:** Escalón unitario.

2.6.6. Escalón unitario

La función escalón unitario o Heaviside está definida por

$$\Theta(t) = \begin{cases} 1 & \text{si } t > 0 \\ 0 & \text{si } t < 0, \end{cases}$$

que es de la forma que se muestra en la figura

Calculemos ahora su transformada de Fourier

$$\begin{aligned} (\mathcal{F}\Theta)(s) &= \int_{-\infty}^{\infty} \Theta(t) e^{-2\pi i st} dt \\ (\mathcal{F}\Theta)(s) &= \int_0^{\infty} e^{-2\pi i st} dt = -\frac{1}{2\pi i s} (e^{-2\pi i st})_0^{\infty} = \frac{1}{2\pi i s} \left(1 - \left(\lim_{t \rightarrow \infty} e^{-2\pi i st} \right) \right). \end{aligned}$$

Esta es un límite complicado de evaluar. ¿Existe una forma más astuta de calcular esta transformada de Fourier? Para esto, veamos una propiedad importante de la función de Heaviside, notemos que

$$\frac{d\Theta}{dt} = \delta(t),$$

2. Transformada de Fourier

ya que por la definición de la delta de Dirac, tenemos que

$$\int_{-\infty}^x \delta(y) dy = \Theta(x).$$

Entonces, para evaluar finalmente la transformada de Fourier de la función Heaviside, haremos un pequeño truco definiendo a la función $f(t) = \Theta(t) - \frac{1}{2}$, y con la propiedad de las derivadas, tenemos que

$$(\mathcal{F}\delta)(s) = (\mathcal{F}f')(s) = 2\pi i s (\mathcal{F}f)(s),$$

entonces tenemos que

$$(\mathcal{F}f)(s) = \frac{(\mathcal{F}\delta)(s)}{2\pi i s} = \frac{1}{2\pi i s},$$

y por otro lado

$$(\mathcal{F}f)(s) = (\mathcal{F}\Theta)(s) - \left(\mathcal{F}\frac{1}{2} \right)(s) = (\mathcal{F}\Theta)(s) - \frac{\delta(t)}{2},$$

y por lo tanto,

$$(\mathcal{F}\Theta)(s) = \frac{\delta(t)}{2} + \frac{1}{2\pi i s}.$$

Parece arbitrario el uso de la función f . Sin embargo, no profundizaremos más en el tema ya que requiere de matemáticas más avanzadas en teoría de distribuciones para poder tener una explicación formal. La gráfica se ve en la figura 2.12.

2.7. Transformada de Fourier en varias dimensiones

Vamos a generalizar la transformada de Fourier a más dimensiones. Vamos a realizar un estudio muy enfocado a dos dimensiones, pero muchas de las propiedades que veremos son generalizables a n dimensiones. Las aplicaciones de la transformada de Fourier en más de una dimensión son muchas. Cambiaremos un poco el lenguaje ya que hablar de tiempo en varias dimensiones no tiene mucho sentido físico, ahora consideraremos a la variable no transformada como una variable espacial. Veremos que las ideas en una dimensión se reflejan aquí. Pongamos en claro las notaciones.

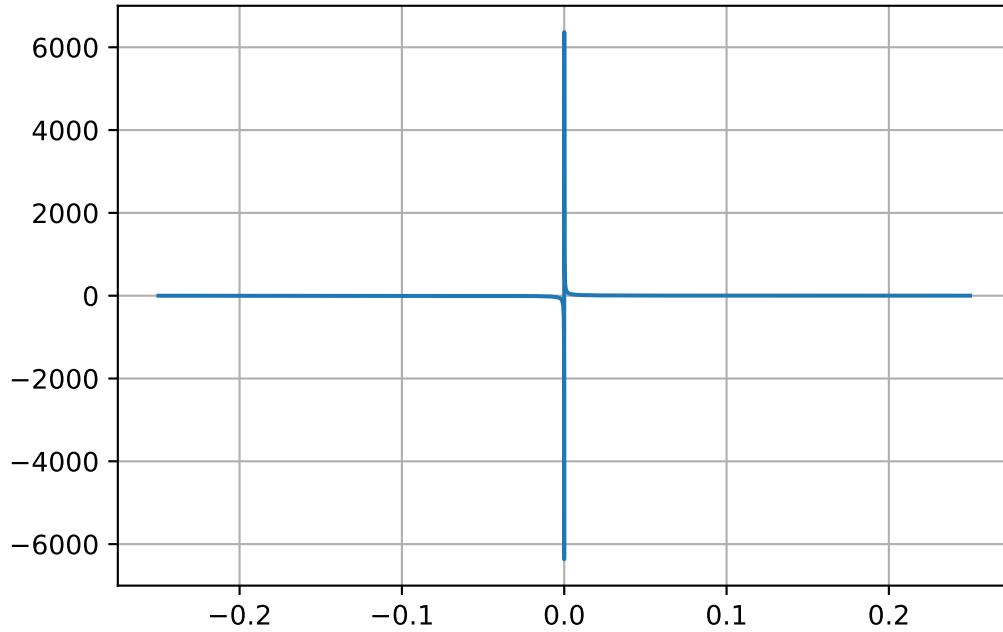
Notación Las variables espaciales serán vectores que serán descritos por letras romanas

$$\vec{x} = (x_1, x_2, \dots, x_n).$$

Las variables en el espacio de frecuencias correspondientes serán vectores descritos por letras griegas

$$\vec{\xi} = (\xi_1, \xi_2, \dots, \xi_n).$$

Ya hemos estudiado las series de Fourier en varias dimensiones. Ahora veamos a la transformada de Fourier.

**Figura 2.12.:** Transformada de Fourier de la función Heaviside.

Definición 2.3. Consideremos entonces a una función f en \mathbb{R}^n . Entonces la transformada de Fourier de f es $\mathcal{F}f$, definida por

$$(\mathcal{F}f)(\vec{\xi}) = \int_{\mathbb{R}^n} e^{-2\pi i \vec{x} \cdot \vec{\xi}} f(x) d\vec{x}.$$

La transformada inversa de Fourier de una función g es $\mathcal{F}^{-1}g$ está definida por

$$(\mathcal{F}^{-1}g)(\vec{x}) = \int_{\mathbb{R}^n} e^{2\pi i \vec{x} \cdot \vec{\xi}} g(\vec{\xi}) d\vec{\xi}.$$

Ya definidas la transformada de Fourier y la transformada inversa de Fourier, podemos establecer algunas propiedades.

2. Transformada de Fourier

2.7.1. Propiedades

Las propiedades de paridad son exactamente las mismas que en el caso unidimensional. Las propiedades de linealidad y de traslación son las mismas también. La propiedad que se vuelve muy interesante es la de cambio de escala o similitud. Vamos a generalizar esto a una transformación lineal realizada sobre la variable \vec{x} y cómo afecta esto a las transformadas de Fourier. Haremos la deducción aquí para dos dimensiones, pero esto es fácilmente generalizable a n dimensiones.

Un cambio de variables lineal en el dominio espacial es de la forma

$$\begin{pmatrix} u_1 \\ u_2 \end{pmatrix} = \underbrace{\begin{pmatrix} a & b \\ c & d \end{pmatrix}}_{\equiv A} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix},$$

o de forma explícita,

$$\begin{aligned} u_1 &= ax_1 + bx_2 \\ u_2 &= cx_1 + dx_2. \end{aligned}$$

Esto podría parecer algo raro que quisiéramos realizar, pero en realidad no lo es. En el caso de edición de imágenes esto puede corresponder a una distorsión completa de la imagen o bien a una simple rotación que se realiza por medio de la multiplicación de una matriz. Dado que la mayoría de veces, es deseable recuperar a la imagen original, vamos a pedir como restricción que $\det(A) \neq 0$.

Notación Vamos a denotar por A^{-T} a la matriz $(A^{-1})^T$, es decir a la matriz transpuesta de la matriz inversa de A . Entonces el teorema de similitud es el siguiente.

Teorema 2.3. *Teorema de similitud para n dimensiones.*

$$\mathcal{F}(f(A\vec{x})) = \frac{1}{|\det A|} \mathcal{F}f\left(A^{-T}\vec{\xi}\right).$$

Demostración. Empecemos analizando el lado izquierdo de la ecuación.

$$(\mathcal{F}(f(A\vec{x})))\left(\vec{\xi}\right) = \int_{\mathbb{R}^n} e^{-2\pi i \vec{x} \cdot \vec{\xi}} f(A\vec{x}) d\vec{x}.$$

Realicemos entonces un cambio de variable $\vec{u} = A\vec{x}$. Entonces

$$\begin{aligned} \int_{\mathbb{R}^n} e^{-2\pi i \vec{x} \cdot \vec{\xi}} f(A\vec{x}) d\vec{x} &= \int_{\mathbb{R}^n} e^{-2\pi i \vec{x} \cdot \vec{\xi}} f(A\vec{x}) \frac{1}{|\det(A)|} |\det(A)| d\vec{x} \\ &= \frac{1}{|\det(A)|} \int_{\mathbb{R}^n} e^{-2\pi i \vec{x} \cdot \vec{\xi}} f(A\vec{x}) |\det(A)| d\vec{x} \\ \int_{\mathbb{R}^n} e^{-2\pi i \vec{x} \cdot \vec{\xi}} f(A\vec{x}) d\vec{x} &= \frac{1}{|\det(A)|} \int_{\mathbb{R}^n} e^{-2\pi i A^{-1}\vec{u} \cdot \vec{\xi}} f(\vec{u}) d\vec{u}, \end{aligned}$$

recordando que $d\vec{u} = |\det(A)| d\vec{x}$. Ahora recurrimos a la identidad, cuando realizamos el producto punto entre un vector y una matriz multiplicando a un vector

$$\vec{\xi} \cdot B\vec{u} = B^T \vec{\xi} \cdot \vec{u}.$$

Ahora si $B = A^{-1}$, entonces, sustituyendo, tenemos

$$\frac{1}{|\det(A)|} \int_{\mathbb{R}^n} e^{-2\pi i A^{-1}\vec{u} \cdot \vec{\xi}} f(\vec{u}) d\vec{u} = \frac{1}{|\det(A)|} \int_{\mathbb{R}^n} e^{-2\pi i \vec{u} \cdot A^{-T} \vec{\xi}} f(\vec{u}) d\vec{u},$$

que es exactamente lo que buscamos:

$$\mathcal{F}(f(A\vec{x})) = \frac{1}{|\det A|} \mathcal{F}f(A^{-T} \vec{\xi}).$$

□

2.7.2. Algunos ejemplos

Existen algunas funciones $f(x_1, x_2, \dots, x_n)$ de n variables que pueden escribirse como el producto de funciones de una variable

$$f(x_1, x_2, \dots, x_n) = f_1(x_1) f_2(x_2) \cdots f_n(x_n).$$

Cuando una función puede ser factorizada de esta manera, es posible calcular su transformada de Fourier como el producto de sus transformadas de Fourier. Veamos el caso de $n = 2$

$$\begin{aligned} (\mathcal{F}f)(\xi_1, \xi_2) &= \int_{\mathbb{R}^2} e^{-2\pi i \vec{x} \cdot \vec{\xi}} f(\vec{x}) d\vec{x} \\ &= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} e^{-2\pi i (x_1 \xi_1 + x_2 \xi_2)} f(x_1, x_2) dx_1 dx_2 \\ &= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} e^{-2\pi i x_1 \xi_1} e^{-2\pi i x_2 \xi_2} f_1(x_1) f_2(x_2) dx_1 dx_2 \\ &= \underbrace{\int_{-\infty}^{\infty} \left(\int_{-\infty}^{\infty} e^{-2\pi i x_1 \xi_1} f_1(x_1) dx_1 \right) e^{-2\pi i x_2 \xi_2} f_2(x_2) dx_2}_{=(\mathcal{F}f_1)(\xi_1)} \\ &= (\mathcal{F}f_1)(\xi_1) \int_{-\infty}^{\infty} e^{-2\pi i x_2 \xi_2} f_2(x_2) dx_2 \\ (\mathcal{F}f)(\xi_1, \xi_2) &= (\mathcal{F}f_1)(\xi_1) (\mathcal{F}f_2)(\xi_2). \end{aligned}$$

En general, la propiedad se escribe

$$(\mathcal{F}f)(\xi_1, \xi_2, \dots, \xi_n) = (\mathcal{F}f_1)(\xi_1) (\mathcal{F}f_2)(\xi_2) \cdots (\mathcal{F}f_n)(\xi_n).$$

2.7.2.1. Función rectángulo en dos dimensiones

La función rectángulo en dos dimensiones es una función que vale 1 en un cuadrado de lado 1 y está centrado en el origen. Matemáticamente se define como

$$\Pi(x_1, x_2) = \begin{cases} 1 & -\frac{1}{2} < x_1 < \frac{1}{2}, -\frac{1}{2} < x_2 < \frac{1}{2} \\ 0 & \text{en cualquier otro caso.} \end{cases}$$

La gráfica de esta función se ve en la figura 2.13.

2. Transformada de Fourier

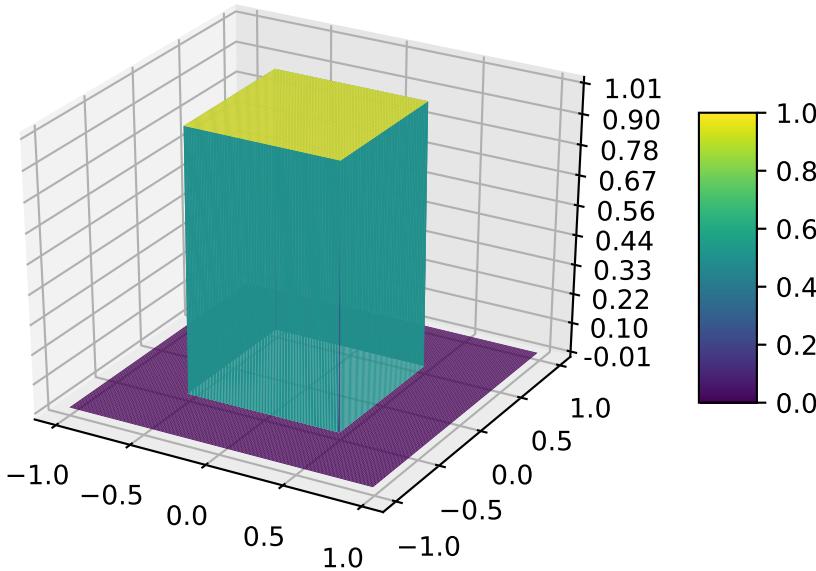


Figura 2.13.: Función rectángulo en dos dimensiones.

Es posible factorizar a $\Pi(x_1, x_2)$ como un producto de dos funciones rectángulo en una dimensión

$$\Pi(x_1, x_2) = \Pi(x_1) \Pi(x_2).$$

Entonces calcular la transformada de Fourier es sencillo y tenemos que

$$(\mathcal{F}\Pi)(\xi_1, \xi_2) = \text{sinc}(\xi_1) \text{sinc}(\xi_2).$$

La gráfica se muestra en la figura 2.14.

2.7.2.2. Función gaussiana

Otro buen ejemplo es el de la gaussiana. La función gaussiana en dos dimensiones está definida por

$$g(\vec{x}) = e^{-\pi|x|^2}.$$

La gráfica puede verse entonces en la figura 2.15. Entonces, así como en el caso unidimen-

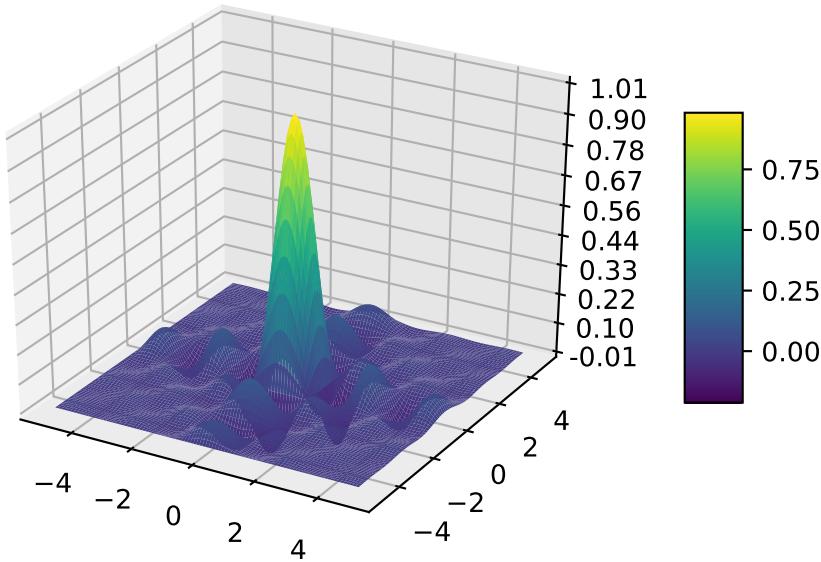


Figura 2.14.: Transformada de Fourier de la función rectángulo en dos dimensiones.

sional, la transformada de Fourier de la función gaussiana es ella misma. Entonces

$$(\mathcal{F}g)(\vec{\xi}) = e^{-\pi|\vec{\xi}|^2}.$$

2.7.3. Coordenadas polares

En los curso de cálculo, una discusión que inicia tarde o temprano después de generalizar a varias dimensiones es el de cambios de coordenadas. Esto se debe a que muchas veces, parece más natural utilizar otros sistemas de coordenadas que no sean los usuales rectangulares o cartesianos. Vamos entonces a concentrarnos en las transformadas de Fourier en coordenadas polares y de forma más específica en funciones que se conocen como radiales.

Definición 2.4. Una función es radial si depende solo de la distancia desde el origen^a. Esto quiere decir que solo dependen de r y no de θ , utilizando la notación usual de coordenadas polares (r, θ) .

^aTambién se les conoce como isotrópicas.

2. Transformada de Fourier

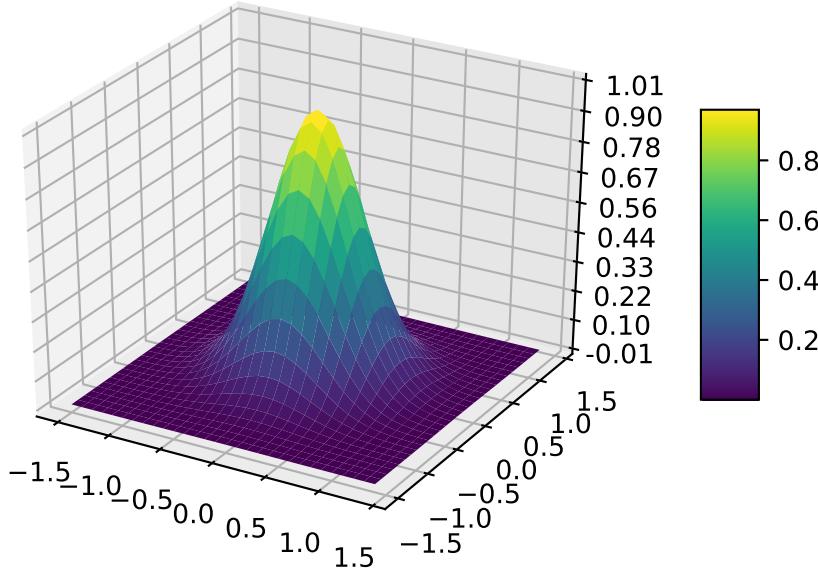


Figura 2.15.: Función gaussiana en dos dimensiones.

Veamos entonces cómo se ve la transformada de Fourier para una función en coordenadas polares. Debemos convertir entonces a

$$\int_{\mathbb{R}^2} e^{-2\pi i \vec{x} \cdot \vec{\xi}} f(\vec{x}) d\vec{x} = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} e^{-2\pi(x_1 \xi_1 + x_2 \xi_2)} f(x_1, x_2) dx_1 dx_2$$

a coordenadas polares. Vamos a asumir que $f(\vec{x})$ es una función radial, lo que significa que

$$f(\vec{x}) = f(r).$$

Los límites de integración para r son desde 0 hasta ∞ y para θ son desde 0 hasta 2π .

La medida de integración $dx_1 dx_2$ se vuelve

$$r dr d\theta.$$

Por último, debemos ver cómo se traduce el producto punto $\vec{x} \cdot \vec{\xi} = x_1 \xi_1 + x_2 \xi_2$ en la exponencial. Para esto vamos a hacer la correspondencia

$$\begin{aligned} (x_1, x_2) &= (r, \theta) \\ (\xi_1, \xi_2) &= (\rho, \phi), \end{aligned}$$

entonces

$$\vec{x} \cdot \vec{\xi} = [\vec{x}] [\vec{\xi}] \cos(\theta - \phi) = r\rho \cos(\theta - \phi).$$

La transformada de Fourier es entonces

$$\int_{\mathbb{R}^2} e^{-2\pi i \vec{x} \cdot \vec{\xi}} f(\vec{x}) d\vec{x} = \int_0^{2\pi} \int_0^\infty f(r) e^{-2\pi i r \rho \cos(\theta - \phi)} r dr d\theta.$$

Ahora, ¿cómo lidiamos con esta integral? Primero, analicemos la exponencial. Esta es periódica en θ con período θ y entonces la integral de

$$\int_0^{2\pi} e^{-2\pi i r \rho \cos(\theta - \phi)} d\theta$$

no depende de ϕ ¹. Por lo tanto

$$\int_0^{2\pi} e^{-2\pi i r \rho \cos(\theta - \phi)} d\theta = \int_0^{2\pi} e^{-2\pi i r \rho \cos(\theta)} d\theta.$$

Esta es una función conocida. Tiene la forma de la función de Bessel de orden cero del primer tipo. Esta está definida, en general, como

$$J_0(a) = \frac{1}{2\pi} \int_0^{2\pi} e^{-ia \cos(\theta)} d\theta.$$

De manera que

$$\int_0^{2\pi} e^{-2\pi i r \rho \cos(\theta)} d\theta = 2\pi J_0(2\pi r \rho),$$

y finalmente, la transformada de Fourier de $f(r)$ es entonces

$$(\mathcal{F}f)(\rho) = 2\pi \int_0^\infty f(r) J_0(2\pi r \rho) r dr.$$

Notemos que también tenemos aquí un teorema.

Teorema 2.4. *La transformada de Fourier de una función radial también es radial.*

¹En efecto, si una función $g(\theta, \phi)$ es periódica en θ de 2π , entonces $G(\phi) = \int_0^{2\pi} g(\theta - \phi) d\theta$, entonces

$$\begin{aligned} G'(\phi) &= \int_0^{2\pi} \frac{dg(\theta - \phi)}{d\phi} d\theta \\ &= \int_0^{2\pi} \frac{dg(\theta - \phi)}{d(\theta - \phi)} \frac{d(\theta - \phi)}{d\phi} d\theta \\ &= - \int_0^{2\pi} \frac{dg(\theta - \phi)}{d(\theta - \phi)} d\theta \quad \text{pero } \frac{dg(\theta - \phi)}{d\theta} = \frac{dg(\theta - \phi)}{d(\theta - \phi)} \frac{d(\theta - \phi)}{d\theta} = \frac{dg(\theta - \phi)}{d\theta} \\ &= - \int_0^{2\pi} dg(\theta - \phi) d\theta \\ &= - \int_0^{2\pi} dg(\theta - \phi) \\ G'(\phi) &= - g(\theta - \phi)|_{\theta=0} = 0 \quad \text{porque } g \text{ es periódica en } \theta. \end{aligned}$$

Entonces $G(\phi)$ es constante y no depende de ϕ .

2. Transformada de Fourier

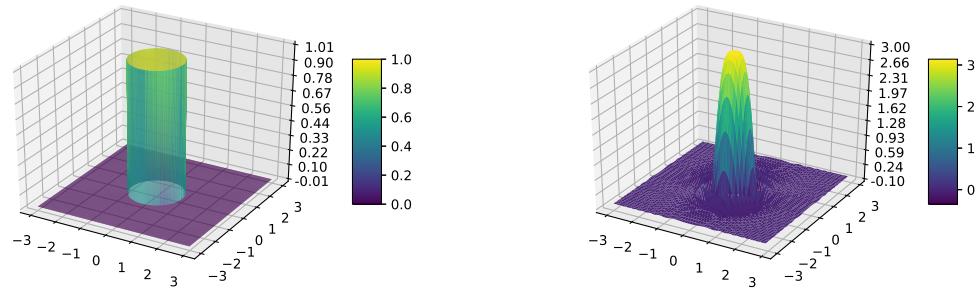


Figura 2.16.: Función circ del lado izquierdo y su transformada de Fourier del lado derecho.

2.7.3.1. Funciones de Bessel

No profundizaremos sobre las funciones de Bessel. Si hay interés por esto, los libros de métodos matemáticos para la física tienen una amplia explicación del tema. Solo vamos a contentarnos con una de las definiciones de las funciones de Bessel de orden n del primer tipo:

$$J_n(a) = a^n \sum_{m=0}^{\infty} \frac{(-1)^m a^{2m}}{2^{2m+n} m! (n+m)!}.$$

También vamos a establecer las relaciones de recurrencia

$$\frac{d}{dx} (a^n J_n(a)) = a^n J_{n-1}(a).$$

2.7.3.2. Ejemplo: funciones circ y jinc

Una función radial interesante que es como la versión radial de la función rectángulo es

$$c(r) = \begin{cases} 1 & \text{si } r < 1 \\ 0 & \text{si } r \geq 1. \end{cases}$$

La gráfica puede verse en la figura 2.16 del lado izquierdo.

Para la transformada de Fourier de circ, tomamos la definición y entonces

$$(\mathcal{F}c)(\rho) = 2\pi \int_0^\infty c(r) J_0(2\pi r\rho) r dr = 2\pi \int_0^1 J_0(2\pi r\rho) r dr.$$

2.7. Transformada de Fourier en varias dimensiones

Ahora utilizamos las propiedades de las funciones de Bessel que vimos en la sección anterior, y tenemos entonces que

$$(\mathcal{F}c)(\rho) = \frac{J_1(2\pi)}{\rho}.$$

Si definimos ahora la función jinc que es de la forma

$$j(\rho) = \frac{J_1(\pi\rho)}{2\rho},$$

entonces

$$(\mathcal{F}c)(\rho) = 4\text{jinc}(2\rho).$$

La gráfica de esta función puede verse en la figura 2.16 del lado derecho.

3. Convolución y filtros

3.1. Motivación, definiciones y teorema de convolución

La motivación de este capítulo es que ya sabemos interpretar y analizar señales entonces procede investigar cómo podemos modificar a las señales. Algunas de las propiedades de la transformada de Fourier van en esta dirección. por ejemplo, podemos sumar a dos señales y su transformada de Fourier será la suma de las transformadas individuales:

$$\mathcal{F}(f + g) = \mathcal{F}f + \mathcal{F}g.$$

Esta es una forma sencilla de complicar o simplificar a una señal. La siguiente operación binaria que podemos considerar es la multiplicación. El caso sencillo es el de multiplicar por una constante, teniendo que

$$\mathcal{F}(af) = a\mathcal{F}f.$$

Entonces el espectro es afectado por el mismo factor que en el espacio de las frecuencias que en el espacio del tiempo.

Si queremos generalizar un poco más, ¿existe una combinación de las señales $f(t)$ y $g(t)$ tal que la transformada de Fourier sea

$$(\mathcal{F}g)(s)(\mathcal{F}f)(s)?$$

Analicemos esta posibilidad. Calculemos la expresión, utilizando variables de integración distintas (truco similar al del cálculo de la integral de la distribución gaussiana),

$$\begin{aligned} (\mathcal{F}g)(s)(\mathcal{F}f)(s) &= \int_{-\infty}^{\infty} e^{-2\pi ist} g(t) dt \int_{-\infty}^{\infty} e^{-2\pi isx} f(x) dx \\ &= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} e^{-2\pi ist} e^{-2\pi isx} g(t) f(x) dt dx \\ (\mathcal{F}g)(s)(\mathcal{F}f)(s) &= \int_{-\infty}^{\infty} \left(\int_{-\infty}^{\infty} e^{-2\pi i s(t+x)} g(t) dt \right) f(x) dx. \end{aligned}$$

Ahora proponemos un cambio de variables $u = t + x$ y entonces

$$\begin{aligned} (\mathcal{F}g)(s)(\mathcal{F}f)(s) &= \int_{-\infty}^{\infty} \left(\int_{-\infty}^{\infty} e^{-2\pi i su} g(u-x) du \right) f(x) dx \\ &= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} e^{-2\pi i su} g(u-x) f(x) dudx \\ &= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} e^{-2\pi i su} g(u-x) f(x) dx du \\ &= \int_{-\infty}^{\infty} e^{-2\pi i su} \underbrace{\left(\int_{-\infty}^{\infty} g(u-x) f(x) dx \right)}_{\equiv h(u)} du \\ (\mathcal{F}g)(s)(\mathcal{F}f)(s) &= \mathcal{F}h(s). \end{aligned}$$

3. Convolución y filtros

Regresando a las variables originales,

$$h(t) = \int_{-\infty}^{\infty} g(t-x) f(x) dx.$$

Hemos resuelto la pregunta, por lo que estamos listos para definir a la convolución.

Definición 3.1. La convolución de dos funciones $g(t)$ y $f(t)$ es la función

$$h(t) = \int_{-\infty}^{\infty} g(t-x) f(x) dx.$$

Vamos a utilizar entonces la notación, para expresar en términos de f y g :

$$(g * f)(t) = \int_{-\infty}^{\infty} g(t-x) f(x) dx.$$

Y entonces con la deducción de la definición de convolución, hemos probado el teorema de convolución que dicta lo siguiente.

Teorema 3.1. *Teorema de convolución. Una convolución en el dominio del tiempo es un producto en el dominio de las frecuencias.*

$$(\mathcal{F}(g * f))(s) = (\mathcal{F}g)(s) (\mathcal{F}f)(s).$$

Es importante realizar una pausa para anotar algunas confusiones que pueden surgir por usar una mala notación. El hecho de utilizar a dos variables x y t para ver el producto de las transformadas de Fourier es más que un truco matemático. Es realmente importante. Veamos por qué. Supongamos que utilizamos la misma variable y queremos calcular $g(2t) * f(t)$. ¿Qué interpretación es la correcta?

$$g(2t) * f(t) = \int_{-\infty}^{\infty} g(2t-x) f(x) dx \text{ o bien } g(2t) * f(t) = \int_{-\infty}^{\infty} g(2(t-x)) f(x) dx.$$

Analicémoslo. Definimos a $h(t) = g(2t)$, entonces

$$(h * f)(t) = \int_{-\infty}^{\infty} h(t-x) f(x) dx = \int_{-\infty}^{\infty} g(2(t-x)) f(x) dx.$$

Entonces la elección correcta es la segunda.

Ejemplo 3.1. Veamos un ejemplo. Calculemos la convolución de la función rectángulo consigo misma

$$(\Pi * \Pi)(x) = \int_{-\infty}^{\infty} \Pi(x-t) \Pi(t) dt$$

Recordemos que

$$\Pi(t) = \begin{cases} 1 & |t| < \frac{1}{2} \\ 0 & |t| > \frac{1}{2}. \end{cases}$$

Entonces

$$\Pi(x-t) = \begin{cases} 1 & |x-t| < \frac{1}{2} \\ 0 & |x-t| > \frac{1}{2}. \end{cases}$$

Por lo tanto, los límites de integración que debemos considerar en los que $\Pi(x-t)\Pi(t) \neq 0$, entonces, veamos cuándo $\Pi(x-t) \neq 0$ y esto es en el intervalo entre $x - \frac{1}{2}$ y $x + \frac{1}{2}$ y al mismo tiempo, en el intervalo entre $-\frac{1}{2}$ y $\frac{1}{2}$. También, x debe estar en el intervalo $[-1, 1]$.

Consideraremos el primer caso. Si $-1 < x < 0$, tenemos que $x - \frac{1}{2} < -\frac{1}{2}$ y $x + \frac{1}{2} < \frac{1}{2}$, entonces el intervalo en el que $\Pi(x-t)\Pi(t) \neq 0$ es $[-\frac{1}{2}, x + \frac{1}{2}]$, por lo que

$$(\Pi * \Pi)(x) = \int_{-1/2}^{x+1/2} dt = t \Big|_{-1/2}^{x+1/2} = x + 1.$$

Ahora consideraremos el caso en el que $x > 0$, entonces tenemos que $x - \frac{1}{2} > -\frac{1}{2}$ y $x + \frac{1}{2} > \frac{1}{2}$, por lo que el intervalo de integración es $[x - \frac{1}{2}, \frac{1}{2}]$, por lo que

$$(\Pi * \Pi)(x) = \int_{x-1/2}^{1/2} dt = t \Big|_{x-1/2}^{1/2} = 1 - x.$$

Resumiendo, tenemos que

$$(\Pi * \Pi)(x) = \begin{cases} 1 - |x| & \text{si } |x| \leq 1 \\ 0 & \text{en cualquier otro caso,} \end{cases}$$

que es una función que hemos estudiado antes, la función triángulo $\Lambda(x)$, cuya transformada de Fourier es $\text{sinc}(x)$. Utilizando el teorema de convolución, sabiendo que $(\mathcal{F}\Pi)(s) = \text{sinc}(s)$, tenemos que

$$(\mathcal{F}(\Pi * \Pi))(s) = (\mathcal{F}\Pi)(s)(\mathcal{F}\Pi)(s) = \text{sinc}^2(s).$$

3.2. Convoluciones y transformada inversa

Si analizamos el teorema de convolución, y por las simetrías que hay entre la transformada de Fourier y la transformada inversa de Fourier, es sencillo deducir que

$$\mathcal{F}^{-1}(g * f) = \mathcal{F}^{-1}g \cdot \mathcal{F}^{-1}f.$$

Aunque, más interesante es el teorema siguiente

Teorema 3.2. *Teorema de convolución segunda versión. Una multiplicación en el dominio de del tiempo corresponde a una convolución en el dominio de las frecuencias*

$$(\mathcal{F}(gf))(s) = (\mathcal{F}g * \mathcal{F}f)(s).$$

3. Convolución y filtros

Demostración. Recordemos que en las fórmulas de dualidad de las transformadas de Fourier, tenemos que

$$(\mathcal{F}(\mathcal{F}f))(s) = f(-s),$$

entonces para derivar la identidad $\mathcal{F}(gf) = \mathcal{F}g * \mathcal{F}f$, vamos a definir a $h \equiv \mathcal{F}f$ y $ak \equiv \mathcal{F}g$, para demostrar que $\mathcal{F}(gf) = k * h$. Recordemos entonces que $\mathcal{F}(k * h) = (\mathcal{F}k)(\mathcal{F}h)$, por el teorema de convolución. Pero también sabemos que $\mathcal{F}k = \mathcal{F}\mathcal{F}g$ y por propiedades de dualidad, $\mathcal{F}k = g^-$ y de la misma manera, $\mathcal{F}h = f^-$. Por lo que $\mathcal{F}(k * h) = g^- f^- = (gf)^-$, o bien

$$gf = \mathcal{F}(k * h)^-,$$

por lo tanto, si realizamos la transformada de Fourier de ambos lados, tenemos que

$$\mathcal{F}(gf) = \mathcal{F}(\mathcal{F}(k * h)^-) = k * h = \mathcal{F}g * \mathcal{F}f.$$

□

3.3. Interpretación de la convolución

La mejor forma de interpretar un nuevo concepto es estudiando lo que hace. Analicemos si podemos visualizar una convolución. No es una tarea sencilla, pero probemos. Vamos a empezar por analizar la expresión

$$(g * f)(t) = \int_{-\infty}^{\infty} g(t-x) f(x) dx.$$

1. Entonces, fijemos un valor de t y la gráfica de $g(x-t)$ no es más que la gráfica de $g(x)$ pero trasladada a la derecha por un valor t , vamos a realizar este ejemplo con la función sierra para g y la función sinc para f . Ilustramos esto en la figura 3.1. Esto se ve en la transición entre la primera y la segunda gráfica de la figura.
2. Después de esto, para visualizar $g(t-x)$, debemos realizar una reflexión respecto al eje de las ordenadas, tal y como se ve en la figura 3.1. Esto se ve en la transición entre la segunda y la tercera gráfica de la figura.
3. Ahora, multipliquemos el resultado por f que es la función sinc. El resultado se ve en la figura 3.1. Esto se ve en la transición entre la tercera y la cuarta gráfica de la figura.
4. Y finalmente, hay que integrarla, variando t . Esta es la parte realmente complicada de visualizar.

Es difícil realizar la última interpretación gráfica de la convolución, la integración, dejando a t como variable, pero nos sirve este análisis para comprender mejor la interpretación que haremos a continuación.

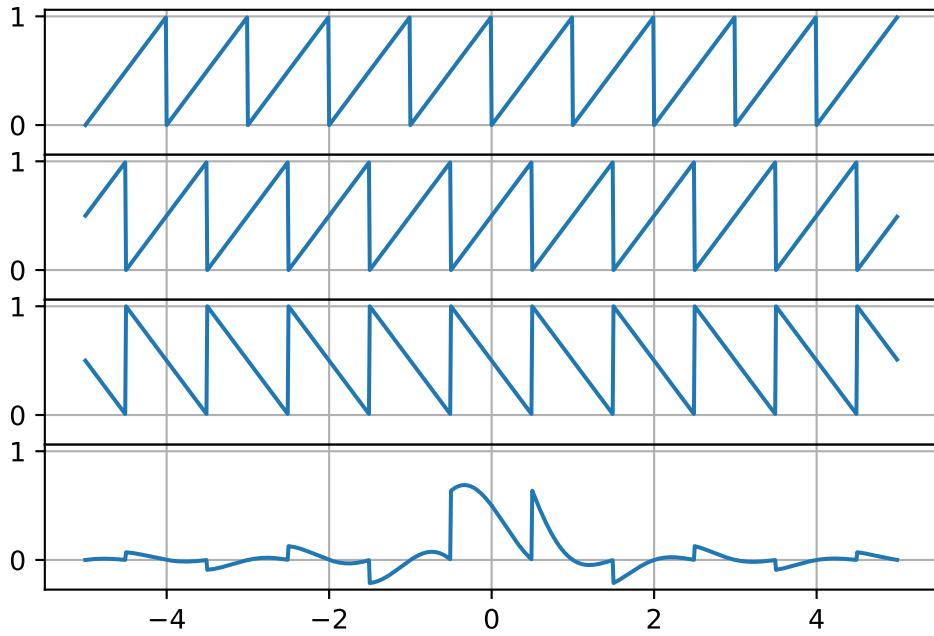


Figura 3.1.: Función sierra trasladada, función sierra trasladada por un factor $t = 0,5$, función sierra trasladada y reflejada y por último, multiplicada por sinc.

En realidad, la interpretación que debemos hacer es que una función *suaviza y promedia* a la otra: g suaviza a f en $g * f$. Muchas veces, g es positiva y se concentra cerca de 0 y con superficie 1, $\int_{-\infty}^{\infty} g(x) dx = 1$, como una gaussiana muy angosta. La visualización previa de la convolución utilizando la función gaussiana se ve en la figura 3.2. Entonces, como vemos, $g(t-x)$ está concentrada alrededor de t y su área debajo de la curva queda intacta: $\int_{-\infty}^{\infty} g(t-x) dx = 1$. Por lo tanto, la integral

$$\int_{-\infty}^{\infty} g(t-x) f(x) dx,$$

es como un promedio ponderado por g alrededor de $x = t$.

Esta es la parte de promediar: calcular la convolución $g * f$ reemplaza el valor $f(t)$ por un promedio ponderado de f alrededor de t . La parte de suavizar, ya la hemos visto cuando estudiamos la convolución de la función rectángulo consigo misma $\Pi * \Pi$ que toma una función que no es continua y la transforma en una función continua. Ahora llega el

3. Convolución y filtros

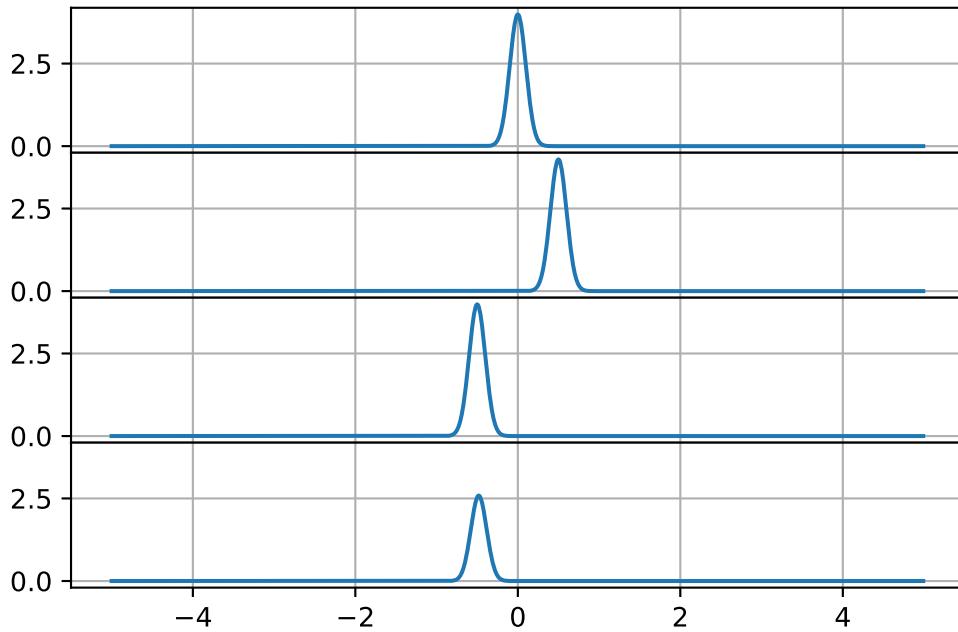


Figura 3.2.: Visualización de la convolución con la función gaussiana.

momento de estudiar un poco las propiedades de la convolución.

3.4. Propiedades de la convolución

En muchos aspectos, la convolución es muy parecida a la multiplicación. Esto tiene mucho sentido, por el teorema de convolución. Veamos algunas de estas.

Proposición 3.1. *La convolución es conmutativa.*

$$f * g = g * f.$$

puede parecer intuitivo si lo vemos a través del teorema de convolución. Sin embargo, parecería contra intuitivo si hacemos el análisis de la interpretación de la convolución, así que probaremos la proposición.

Esto

Demostración. A pesar de lo que nos dicta la intuición, no utilizaremos el teorema de

3.4. Propiedades de la convolución

convolución. Haremos la prueba de la conmutatividad desde la definición

$$\begin{aligned}(f * g)(t) &= \int_{-\infty}^{\infty} f(t-u)g(u)du \\ &= \int_{-\infty}^{\infty} g(t-v)f(v)dv \quad (\text{con } v=t-u) \\ (f * g)(t) &= (g * f)(t).\end{aligned}$$

□

También existe la propiedad de asociatividad para las convoluciones y distributiva.

Proposición 3.2. *La convolución es asociativa.*

$$(f * g) * h = f * (g * h).$$

La convolución es distributiva.

$$f * (g + h) = f * g + f * h.$$

Demostación. De nuevo, partimos de la definición, entonces demostremos primero la asociatividad,

$$\begin{aligned}((f * g) * h)(t) &= \int_{-\infty}^{\infty} (f * g)(t-u)h(u)du \\ &= \int_{-\infty}^{\infty} \left(\int_{-\infty}^{\infty} f(t-u-v)g(v)dv \right) h(u)du \\ &= \int_{-\infty}^{\infty} \left(\int_{-\infty}^{\infty} f(w)g(t-u-w)dw \right) h(u)du \quad \text{con } w=t-u-v \\ &= \int_{-\infty}^{\infty} f(w) \left(\int_{-\infty}^{\infty} g(t-u-w)h(u)du \right) dw \\ &= \int_{-\infty}^{\infty} f(w)(g * h)(t-w)dw \\ &= ((g * h) * f)(t) \\ ((f * g) * h)(t) &= (f * (g * h))(t),\end{aligned}$$

por la propiedad de convolución.

Ahora veamos la distributividad,

$$\begin{aligned}(f * (g + h))(t) &= \int_{-\infty}^{\infty} f(t-u)(g+h)(u)du \\ &= \int_{-\infty}^{\infty} (f(t-u)g(u)) + (f(t-u)h(u))du \\ &= \int_{-\infty}^{\infty} (f(t-u)g(u))du + \int_{-\infty}^{\infty} (f(t-u)h(u))du \\ (f * (g + h))(t) &= (f * g)(t) + (f * h)(t).\end{aligned}$$

□

¿Qué hay del elemento neutro? ¿Existe? Estas preguntas se resumen en buscar una función tal que

$$(g * f)(t) = f(t),$$

para todas las funciones f . Resulta que sí existe.

Proposición 3.3. *El elemento neutro de la convolución es la delta de Dirac.*

3. Convolución y filtros

Demostración. Vamos a utilizar aquí el teorema de convolución. Si buscamos a g tal que

$$g * f = f,$$

entonces

$$\mathcal{F}g \cdot \mathcal{F}f = \mathcal{F}f,$$

es decir que buscamos a g de manera que $\mathcal{F}g = 1$, que es la delta de Dirac, definida en la ecuación 2.2. \square

JCuidado! Existe un neutro, pero encontrar los inversos es una tarea bastante complicada.

Notación Vamos a denotar a la transformada de Fourier de una función f como F . En palabras, vamos a nombrar a la función con una letra minúscula y a su transformada por una mayúscula.

$$F(s) = (\mathcal{F}f)(s).$$

3.5. Aplicaciones con filtros

Sin duda, es una de las aplicaciones más importantes del análisis de Fourier. Aplicar un filtro a una señal, no es más que realizar una operación matemática a una señal. Sin embargo, hay una característica importante que no queremos dejar por un lado: necesitamos que la señal de salida sea reconocible o recuperable después de ser filtrada. La mayoría de filtros modifican la información espectral de la señal y por lo tanto se traducen como una operación en la transformada de Fourier de la señal. Vamos a analizar algunos de los filtros más utilizados a través de la multiplicación en el dominio de las frecuencias, y por lo tanto, a través de la convolución en el dominio del tiempo.

Para estudiar eso entonces, necesitamos una señal de entrada y una señal de salida. La operación realizada en el dominio del tiempo es una convolución con una función $h(t)$

$$w(t) = (h * v)(t) = \int_{-\infty}^{\infty} h(t-x)v(x)dx.$$

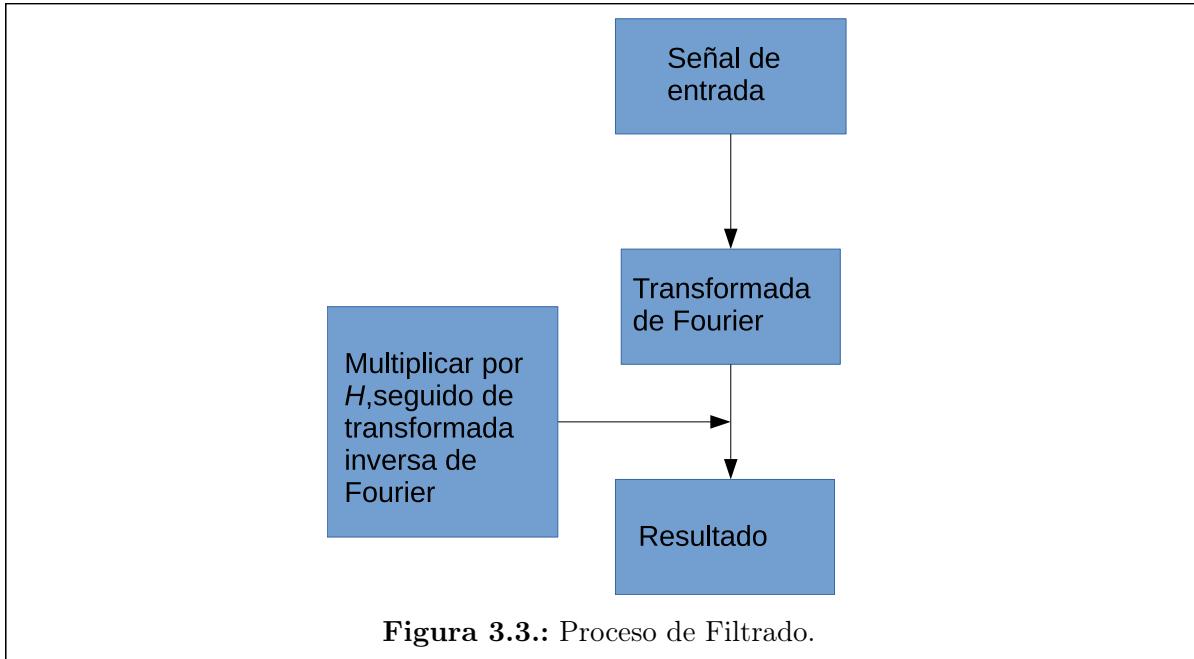
Por lo tanto, por el teorema de convolución, en el dominio de las frecuencias

$$W(s) = H(s)V(s), \quad (3.1)$$

en el contexto de filtros, h se conoce como respuesta de impulso y H se conoce como función de transferencia. Normalmente, diseñamos filtros de manera que una función de transferencia tenga como resultado un efecto específico en una señal de entrada y así diseñamos el filtro a través de la función de transferencia. Entonces el proceso de filtrado de una señal se ve como en la figura 3.3. Veremos algunos ejemplos utilizados.

Pero antes analicemos un poco qué sucede cuando filtramos una señal, qué efecto tiene en su espectro. Partimos de saber que $V(s)$ es una función compleja, por lo que, podemos escribirla como

$$V(s) = |V(s)|e^{i\phi_V(s)} \text{ con } \phi_V(s) = \arctan\left(\frac{\text{Im}(V(s))}{\text{Re}(V(s))}\right),$$



entonces, la fase de $V(s)$ es $\phi_V(s)$ y similarmente, podemos hacer esta descomposición para $W(s)$ y para $H(s)$, de manera que la ecuación 3.1, puede escribirse como

$$\begin{aligned} |W(s)| e^{i\phi_W(s)} &= |H(s)| e^{i\phi_H(s)} |V(s)| e^{i\phi_V(s)} \\ |W(s)| e^{i\phi_W(s)} &= |H(s)| |V(s)| e^{i(\phi_H(s) + \phi_V(s))}. \end{aligned}$$

Entonces, las magnitudes se multiplican y las fases se suman, es decir

$$\begin{aligned} |W(s)| &= |H(s)| |V(s)| \\ \phi_W(s) &= \phi_V(s) + \phi_H(s). \end{aligned}$$

Cuando no tenemos un cambio en la fase, pero sí en la magnitud de la señal, es porque H es una función real. Los filtros más comunes son de este tipo: modifican una parte de las magnitudes del espectro, pero no cambian la fase. Veamos algunos ejemplos.

3.5.1. Pasa bajo

Idealmente, un filtro pasa bajo corta todas las frecuencias mayores a una frecuencia de corte ν_c y permite el paso de las frecuencias menores, sin cambio alguno (por eso se llama pasa bajo). Matemáticamente esto se realiza como tomar la señal original v y realizar una convolución con una función que tenga este efecto en el dominio de las frecuencias:

$$w = h * v,$$

3. Convolución y filtros

o bien, en el dominio de las frecuencias

$$W = H \cdot V.$$

Para lograr el efecto deseado, la función de transferencia ideal es

$$H(s) = \begin{cases} 1 & |s| < \nu_c \\ 0 & |s| \geq \nu_c. \end{cases}$$

Aquí, $H(s)$ no es más que la función rectángulo reescalada:

$$H(s) = \Pi_{2\nu_c}(s) \equiv \Pi\left(\frac{s}{2\nu_c}\right) = \begin{cases} 1 & \left|\frac{s}{2\nu_c}\right| < \frac{1}{2} \\ 0 & \left|\frac{s}{2\nu_c}\right| \geq \frac{1}{2} \end{cases} = \begin{cases} 1 & |s| < \nu_c \\ 0 & |s| \geq \nu_c. \end{cases}$$

Veamos cuál es su transformada inversa,

$$\begin{aligned} h(t) &= \int_{-\infty}^{\infty} H(s) e^{2\pi i s t} ds \\ &= \int_{-\nu_c}^{\nu_c} e^{2\pi i s t} ds \\ &= \frac{e^{2\pi i s t}}{2\pi i t} \Big|_{-\nu_c}^{\nu_c} \\ &= \frac{e^{2\pi i t \nu_c} - e^{-2\pi i t \nu_c}}{2\pi i t} \\ &= \frac{\sin(2\pi \nu_c t)}{\pi t} \\ &= \frac{2\nu_c}{2\nu_c} \frac{\sin(2\pi \nu_c t)}{\pi t} \\ h(t) &= 2\nu_c \text{sinc}(2\nu_c t). \end{aligned}$$

Así pues, hemos logrado reconstruir la función h a partir de H , lo que nos permite terminar el proceso de filtrado. Debemos recuperar a w . Basta con calcular su convolución con v , o bien calcular la transformada de Fourier inversa de W .

3.5.2. Filtros pasa banda

Estos filtros son bastante comunes también. Estos filtros permiten pasar algunas frecuencias sin alterarlas, eliminando otras. El filtro pasa banda tiene como función de transferencia a B y puede ser construido trasladando combinando funciones de transferencia del filtro pasa bajo H .

Centramos nuestro filtro en $\pm\nu_0$ y cortamos las frecuencias ν_c por arriba y por debajo de $\nu_0 \pm \nu_c$, tal y como el pasa bajo elimina frecuencias positivas y negativas. Matemáticamente, B se define como

$$\begin{aligned} B(s) &= \begin{cases} 1 & \nu_0 - \nu_c < |s| < \nu_0 + \nu_c \\ 0 & \text{en cualquier otro caso,} \end{cases} \\ B(s) &= H(s - \nu_0) + H(s + \nu_0). \end{aligned}$$

Podemos visualizar la gráfica de B en la figura 3.4.

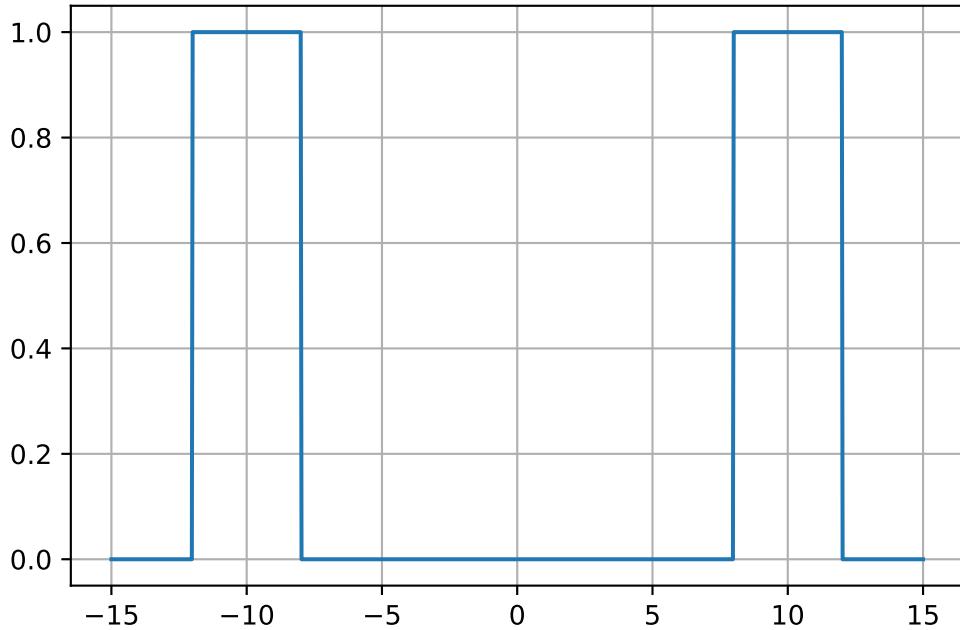


Figura 3.4.: Función de transferencia del filtro pasa banda para $\nu_0 = 10$ y $\nu_c = 2$.

Ahora, al igual que para el filtro pasa bajo, vamos a encontrar la función de respuesta de impulso. Para eso, utilizaremos el teorema de traslación,

$$\begin{aligned} b(t) &= h(t) e^{2\pi i \nu_0 t} + h(t) e^{-2\pi i \nu_0 t} \\ b(t) &= 4\nu_c \cos(2\pi\nu_0 t) \operatorname{sinc}(2\nu_c t). \end{aligned}$$

Su gráfica puede verse en la figura 3.5.

3.5.3. Filtro pasa alto

El filtro pasa alto es otro filtro comúnmente utilizado. Este permite pasar frecuencias por encima de una frecuencia y las demás son eliminadas. Entonces, podemos construir la función de transferencia de este filtro por medio de la función de transferencia del filtro pasa bajo

$$A(s) = 1 - H(s),$$

con ν_c la frecuencia de corte. La gráfica puede verse en la figura

3. Convolución y filtros

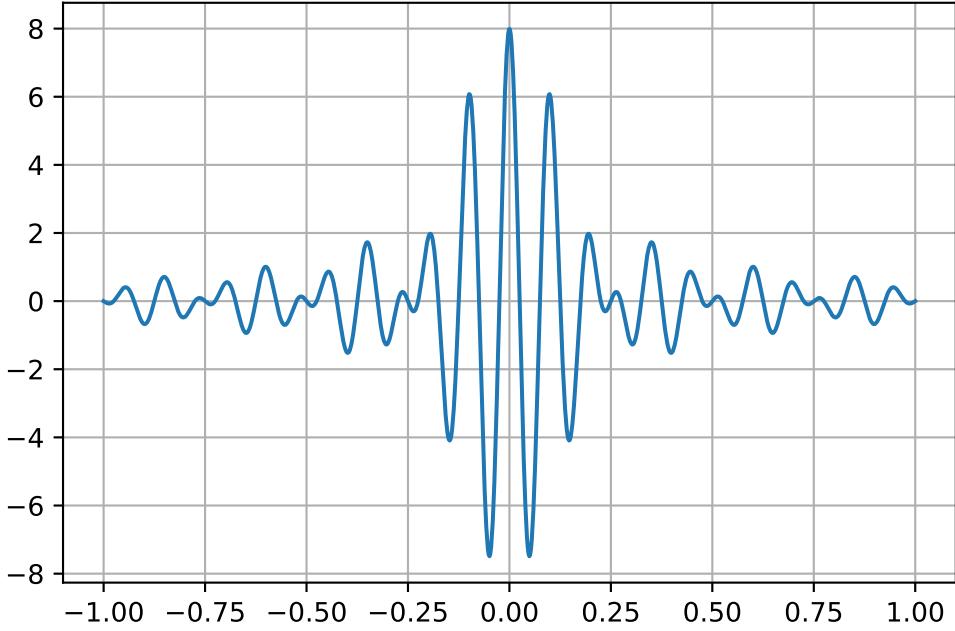


Figura 3.5.: Función de respuesta de impulso del filtro pasa banda para $\nu_0 = 10$ y $\nu_c = 2$.

Veamos cuál es entonces la función de respuesta de impulso a .

$$a(t) = (\mathcal{F}^{-1}(1))(t) - (\mathcal{F}^{-1}H)(t) = \delta(t) - 2\nu_c \text{sinc}(2\nu_c t).$$

Este filtro puede verse como la antípoda del filtro pasa bajo. La gráfica puede verse en la figura 3.7.

3.5.4. Filtro elimina banda

La antípoda entonces del filtro pasa banda es el filtro elimina banda. Este filtro elimina las frecuencias de una banda en particular. La función de transferencia de este filtro es entonces

$$F(s) = 1 - B(s) = 1 - (H(s - \nu_0) + H(s + \nu_0)).$$

Y de igual manera que para el filtro pasa alto, su función de respuesta de impulso es

$$f(t) = (\mathcal{F}^{-1}(1))(t) - (\mathcal{F}^{-1}(H_{-\nu_0} + H_{\nu_0}))(t) = \delta(t) - 2\nu_c \text{sinc}(2\nu_c t),$$

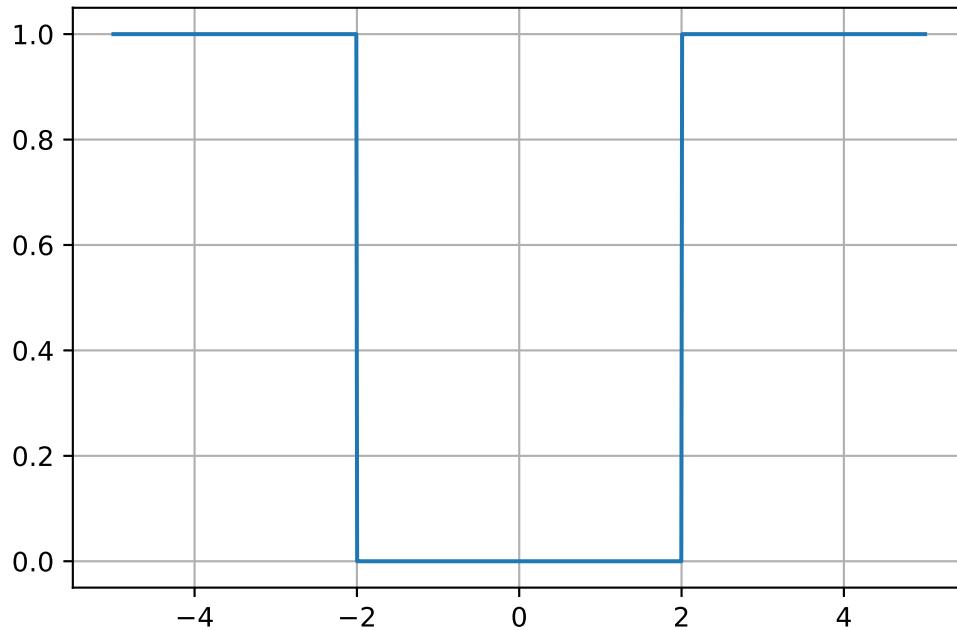


Figura 3.6.: Función de transferencia del filtro pasa alto para $v_c = 2$.

con $H_{\pm\nu_0} = H(s \pm \nu_0)$.

3.6. Generalización a varias dimensiones

La generalización de las convoluciones a varias dimensiones es sencilla

$$(f * g)(\vec{x}) = \int_{\mathbb{R}^n} f(\vec{x} - \vec{y}) g(\vec{y}) d\vec{y}.$$

El teorema de convolución se mantiene.

3. Convolución y filtros

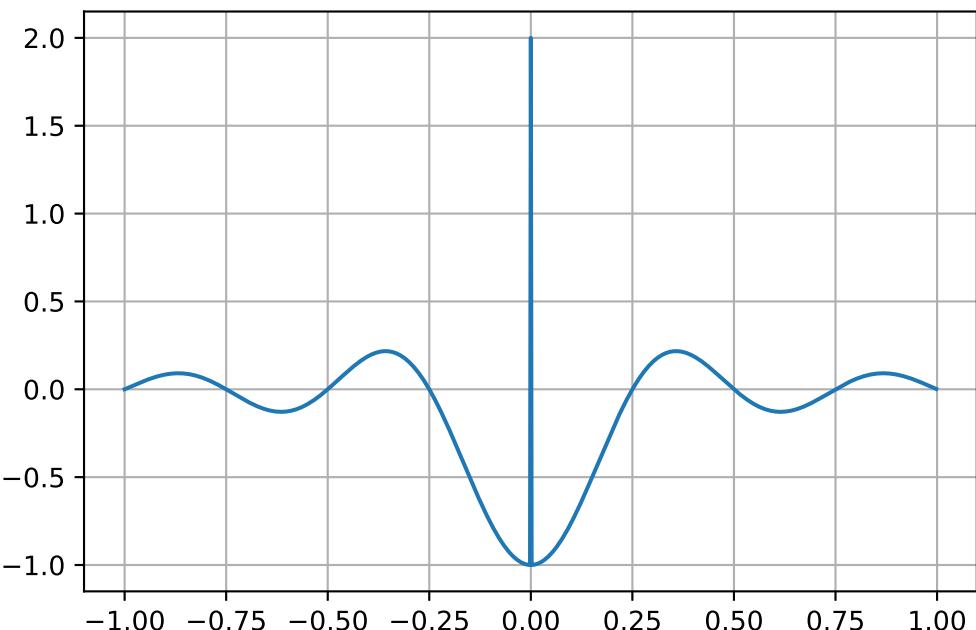


Figura 3.7.: Función de respuesta de impulso del filtro pasa alto para $\nu_c = 2$.

4. Aplicación: tratamiento de imágenes

El tratamiento de imágenes consiste en aplicar transformaciones matemáticas sobre imágenes con el objetivo de mejorar su calidad o de extraer información de la misma. Las aplicaciones de esto son muchas como mejorar la calidad de las fotos, detectar letras o rostros, identificación de zonas cancerosas en física médica, compresión de imágenes, entre muchas otras.

Vamos a exponer las herramientas elementales para el tratamiento de imágenes.

4.1. Imágenes digitales

Antes de tratar las imágenes, debemos saber cómo las interpreta una computadora para poder manipularlas. Vamos a manipular aquí imágenes en dos dimensiones. Una computadora lee un arreglo de dos dimensiones: una matriz en donde cada entrada (llamada pixel) tiene tres valores entre 0 y 255 que corresponden a las intensidades de los colores rojo, verde y azul que salen en la pantalla. Así pues existen varias formas de codificar imágenes, pero esta es la usual. En general, una imagen digital se define de la siguiente manera.

Definición 4.1. Una imagen digital está definida como una función $f(m, n, \dots)$ con valores discretos y finitos cuyas coordenadas m, n, \dots son también discretas y finitas.

$$f : \mathbb{N}^d \rightarrow \mathbb{R}^B.$$

Esto significa que a cada casilla de coordenadas (m, n, \dots) de un arreglo de d enteros, le asigna un vector con B valores reales.

Vamos a tratar imágenes en dos dimensiones, entonces $d = 2$. Como tenemos tres colores en cada pixel, aquí $B = 3$. Si, por ejemplo, $B = 1$, usualmente se toman los valores promedio de los tres de una imagen normal. Como ejemplo, vamos a tomar la imagen¹ de la figura 4.1.

4.2. Primeras manipulaciones

Veamos algunas manipulaciones de una imagen sola que no requiere de otra imagen para poder realizar su manipulación, como convertir una imagen a escala de grises o en blanco y negro o sacar un negativo.

¹Foto tomada desde el edificio Adriatico Guest House. Trieste. Italia.

4. Aplicación: tratamiento de imágenes



Figura 4.1.: Imagen de ejemplo.

4.2.1. Escala de grises

Ahora, tomemos una de las imágenes de la figura 4.1 y hagámosla una imagen en escala de grises. Una de las formas que es sencilla es tomar un promedio de las intensidades de rojo, verde y azul para mostrarlo. El resultado se muestra en la imagen de arriba a la derecha figura 4.2. Podemos ver que la definición de los árboles en la esquina inferior derecha no es del todo satisfactoria. Estudios psicovisuales han determinado que la mejor fórmula no es el promedio sino, la siguiente

$$\text{gris} = 0,2125r + 0,7154g + 0,0721b,$$

donde r, g, b son las intensidades de rojo, verde y azul normalizadas, es decir, divididas por 255. Vamos a trabajar, de ahora en adelante, con las intensidades normalizadas. El resultado de esta operación se ve en la imagen de abajo de la figura 4.2. Podemos apreciar que la definición de los árboles en la esquina inferior derecha es mejor.

El código para realizar esta imagen puede verse en el apéndice A.4.1.

4.2.2. Blanco y negro

A veces, estamos interesados en convertir una imagen a color en una que solo esté en blanco y negro y no en una escala de grises. Entonces, podemos realizar esto de una manera sencilla. Si la intensidad de gris es menor que 0,5, entonces el color desplegado es negro y sino, el color desplegado es blanco. El resultado puede verse en la figura 4.3. Aquí tomamos

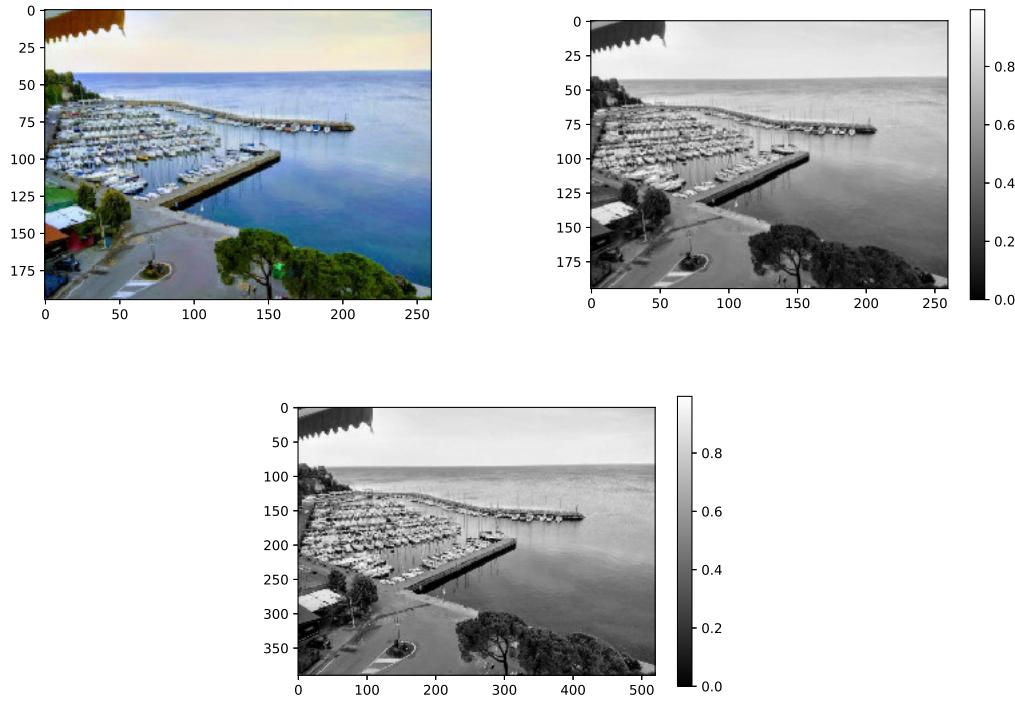


Figura 4.2.: Conversión a grises de una imagen.

como umbral el valor 0,5, sin embargo, podemos cambiarlo. El código para realizar esta transformación puede verse en el apéndice A.4.2.

4.2.3. Negativo

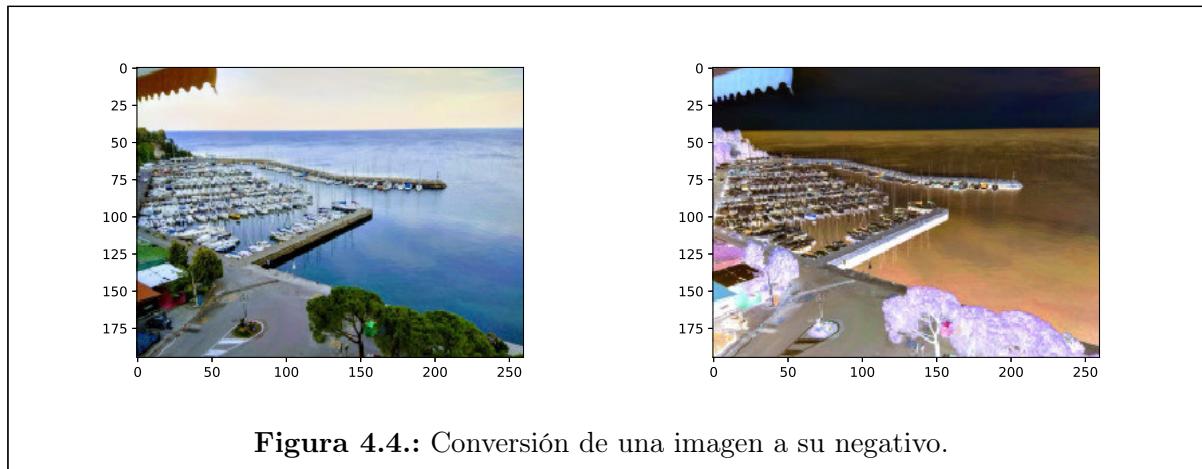
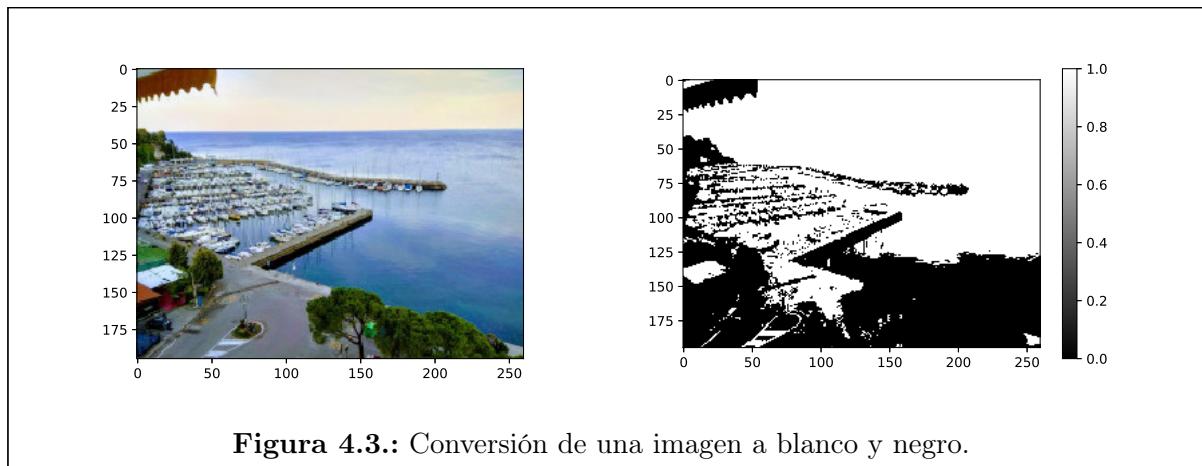
También, puede ser útil o divertido obtener el negativo de una imagen. Para esto utilizamos la función

$$\text{negativo} = 1 - n$$

con $n = r, g, b$, para cada pixel. Esto invierte las intensidades de cada color en cada uno de los pixeles de la imagen.

El resultado puede verse en la figura 4.4. El código para realizar esto está en el apéndice A.4.3.

4. Aplicación: tratamiento de imágenes



4.2.4. Otras manipulaciones

Veamos cómo es esto con las imágenes de grises. Para analizar mejor el efecto, es importante introducir un histograma de intensidades. Si queremos hacer este análisis con una imagen a color, usualmente se presenta un histograma para cada uno de los colores. El histograma mide cuántas veces aparece cada una de las intensidades. También mostraremos la función que aplicamos a cada una de las entradas de los pixeles.

Estos cambios se realizan tomando cada uno de los pixeles y transformándolo por medio de una función

$$j = T(n)$$

donde j es el nuevo valor y n es el valor asignado a cada pixel. Por ejemplo, si deseamos

4.2. Primeras manipulaciones

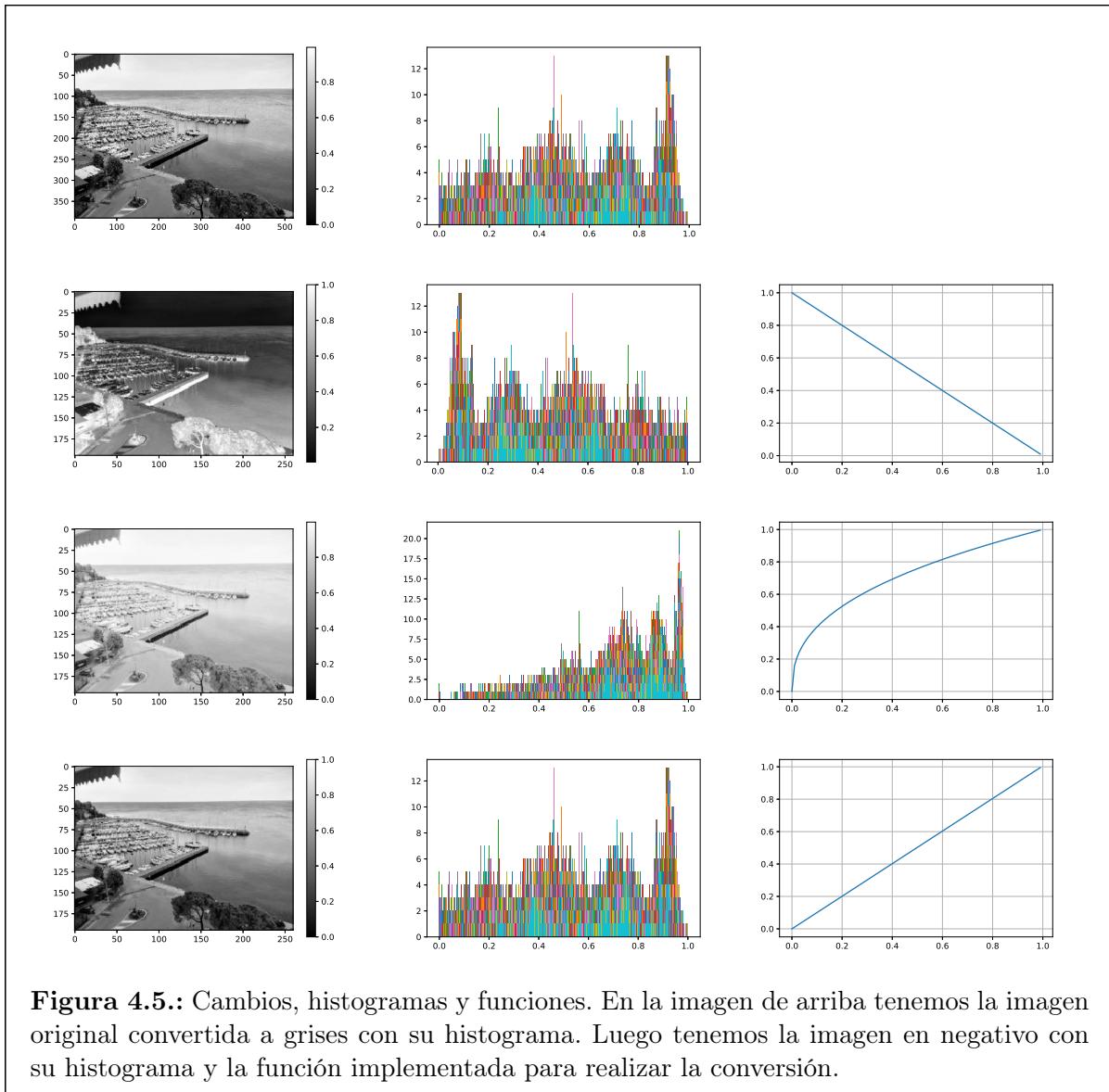


Figura 4.5.: Cambios, histogramas y funciones. En la imagen de arriba tenemos la imagen original convertida a grises con su histograma. Luego tenemos la imagen en negativo con su histograma y la función implementada para realizar la conversión.

realizar una transformación en el negativo, podemos implementar la función

$$T(n) = 1 - n,$$

si deseamos realizar una corrección gamma que modifica la intensidad luminosa, entonces

$$T(n) = n^\gamma.$$

4. Aplicación: tratamiento de imágenes

Para este ejemplo, tomamos a $\gamma = 0.4$. Por último, también podemos realizar una normalización, cuya función es

$$T(n) = \frac{n - n_{\min}}{n_{\max} - n_{\min}}.$$

El efecto de cada uno de estos procesos puede verse en detalle en la figura 4.5. Luego, el código para implementarlo, puede verse en A.4.4.

Para la imagen en negativo, vemos que el histograma está reflejado. Para la corrección gamma, las intensidades bajas disminuyen, lo que produce un efecto de mayor luminosidad en la imagen. Por último, la normalización tiende a estirar horizontalmente al histograma, dándole mejor definición y contraste.

4.3. Transformada de Fourier discreta

Hemos tenido ya una discusión extensa acerca de la transformada de Fourier. Sin embargo, toda ha girado en torno a funciones continuas o al menos continuas por tramos. En las imágenes de una computadora, tenemos funciones que no son continuas, son discretas. ¿Podemos realizar una transformada de Fourier para tales funciones? Resulta que sí. Existe una versión discreta de la transformada de Fourier: la transformada de Fourier discreta. Esta está definida de la siguiente manera.

Definición 4.2. La transformada de Fourier F de una función discreta f está dada por De

$$F(u, v) = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f(m, n) e^{-2\pi i (\frac{um}{M} + \frac{vn}{N})}.$$

Igual manera, es posible implementar la transformada inversa de Fourier discreta.

Definición 4.3. La transformada de Fourier f de una función discreta F está dada por

$$f(m, n) = \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} F(u, v) e^{2\pi i (\frac{um}{M} + \frac{vn}{N})}.$$

Dado que tenemos funciones que son complejas, es necesario utilizar el módulo² de estos números complejos así como su fase o argumento³. No mostraremos aún el resultado de realizar una transformada de Fourier con esta definición. Computacionalmente, esta operación es excesivamente costosa y lenta. Entonces, ¿para qué la estudiamos? Resulta que las transformadas de Fourier discretas son ampliamente utilizadas en tratamiento de imágenes y de señales, en general. Sin embargo, esto es cierto a partir de los años de 1960, cuando se descubrió un algoritmo mucho más eficiente: la transformada rápida de Fourier, cuando Cooley y Tukey presentaron su famoso algoritmo.

² $|z| = \sqrt{\operatorname{Re}(z)^2 + \operatorname{Im}(z)^2}$

³ $\arg(z) = \arctan\left(\frac{\operatorname{Im}(z)}{\operatorname{Re}(z)}\right)$

4.4. Transformada rápida de Fourier

Veamos primero el caso en una dimensión de la transformada rápida de Fourier, ya que esta se implementa para la de dos dimensiones. La propiedad fundamental de las transformadas de Fourier para mejorar el rendimiento es la separabilidad. Así que veamos cómo funciona.

4.4.1. Transformada discreta de Fourier en una dimensión: otra perspectiva

En la sección anterior, la transformada discreta de Fourier está presentada como una discretización de la continua. Casi como la realización de un muestreo del caso continuo. Para poder entenderlo mejor, en una dimensión

$$F(u) = \sum_{n=0}^{N-1} f(n) e^{-\frac{2\pi i u n}{N}}.$$

Ahora, cómo podemos ver esto de forma más computacional. Entonces vamos a trabajar con arreglos de números como

$$\mathbf{f} = (f[0], f[1], \dots, f[N-1]).$$

Entonces utilizando esta convención, la transformada de Fourier discreta se puede definir como a continuación.

Definición 4.4. Sea $\mathbf{f} = (f[0], f[1], \dots, f[N-1])$ un arreglo con N entradas. La transformada discreta de Fourier, $\mathbf{F} = (F[0], F[1], \dots, F[N-1])$ de \mathbf{f} se define como

$$F[m] = \sum_{n=0}^{N-1} f[n] e^{-\frac{2\pi i m n}{N}}.$$

4.4.1.1. Notaciones

Para simplificar las cosas, vamos a establecer algunas notaciones, para esta parte del curso. Parecerán poco ortodoxas para matemáticos, pero para nuestros fines son realmente útiles. Para esto vamos a considerar entonces a dos arreglos

$$\mathbf{x} = (x[0], x[1], \dots, x[N-1]), \mathbf{y} = (y[0], y[1], \dots, y[N-1]),$$

entonces establecemos

$$\begin{aligned} \mathbf{xy} &= (x[0]y[0], x[1]y[1], \dots, x[N-1]y[N-1]) \\ \frac{\mathbf{x}}{\mathbf{y}} &= \left(\frac{x[0]}{y[0]}, \frac{x[1]}{y[1]}, \dots, \frac{x[N-1]}{y[N-1]} \right) \\ \mathbf{x}^p &= (x[0]^p, x[1]^p, \dots, x[N-1]^p) \\ f(\mathbf{x}) &= (f(x[0]), f(x[1]), \dots, f(x[N-1])). \end{aligned}$$

4. Aplicación: tratamiento de imágenes

También, en vamos a utilizar la notación $\mathbf{x}[r : s]$ para describir al arreglo $(\mathbf{x}[r], \mathbf{x}[r+1], \dots, \mathbf{x}[s-1], \mathbf{x}[s])$ y el arreglo $\vec{0}$ será

$$\vec{0} = (0, 0, \dots, 0),$$

el vector cero.

Otra función importante es el vector exponencial complejo, por la definición que hemos visto de la transformada de Fourier discreta. Vamos a establecer que

$$\omega_N = e^{\frac{2\pi i}{N}},$$

y también tenemos que

$$\operatorname{Re}(\omega_N) = \cos\left(\frac{2\pi}{N}\right), \quad \operatorname{Im}(\omega_N) = \sin\left(\frac{2\pi}{N}\right).$$

Entonces, siguiendo la notación que acabamos de definir,

$$\omega_N^N = e^{\frac{2\pi i N}{N}} = 1.$$

De la misma manera

$$\omega_N^{Nn} = 1,$$

y en general, para enteros n y k , tenemos la identidad

$$\omega_N^{Nn+k} = \omega_N^{Nn} \omega_N^k = \omega_N^k.$$

Podemos ver la propiedad también de si N es par, entonces

$$\omega_N^{N/2} = e^{\frac{2\pi i N}{2N}} = e^{i\pi} = -1 \text{ y en general } \omega_N^{\frac{kN}{2}} = (-1)^k.$$

Si deseamos hacer ahora una arreglo de exponentiales complejas, vamos a definir entonces a

$$\vec{\omega} = (1, \omega, \omega^2, \dots, \omega^{N-1}),$$

por último, el vector

$$\vec{1} = (1, 1, \dots, 1).$$

4.4.1.2. Transformada de Fourier discreta para vectores

Con todas estas notaciones, estamos listos para optimizar un poco más nuestra definición de la transformada de Fourier discreta.

Definición 4.5. La transformada de Fourier discreta está dada por

$$F = \sum_{k=0}^{N-1} f[k] \vec{\omega}^{-k}.$$

4.4. Transformada rápida de Fourier

Es importante recordar que la transformada de Fourier de un vector es un vector, de la misma dimensión. Veamos entonces de forma más explícita cómo funciona

$$F[m] = \sum_{k=0}^{N-1} f[k] \vec{\omega}^{-k} [m] = \sum_{k=0}^{N-1} f[k] \omega^{-km}.$$

Ya que tenemos todo en términos de vectores y que la transformada de Fourier es una transformación lineal, debería ser posible escribir a la transformada de Fourier discreta como una matriz. Podemos entonces ver el efecto sobre vectores

$$\begin{pmatrix} F[0] \\ F[1] \\ F[2] \\ \vdots \\ F[N-1] \end{pmatrix} = \begin{pmatrix} \omega^{-0 \cdot 0} & \omega^{-0 \cdot 1} & \omega^{-0 \cdot 2} & \cdots & \omega^{-0 \cdot (N-1)} \\ \omega^{-1 \cdot 0} & \omega^{-1 \cdot 1} & \omega^{-1 \cdot 2} & \cdots & \omega^{-1 \cdot (N-1)} \\ \omega^{-2 \cdot 0} & \omega^{-2 \cdot 1} & \omega^{-2 \cdot 2} & \cdots & \omega^{-2 \cdot (N-1)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \omega^{-(N-1) \cdot 0} & \omega^{-(N-1) \cdot 1} & \omega^{-(N-1) \cdot 2} & \cdots & \omega^{-(N-1) \cdot (N-1)} \end{pmatrix} = \begin{pmatrix} f[0] \\ f[1] \\ f[2] \\ \vdots \\ f[N-1] \end{pmatrix},$$

o bien

$$\begin{pmatrix} F[0] \\ F[1] \\ F[2] \\ \vdots \\ F[N-1] \end{pmatrix} = \begin{pmatrix} 1 & 1 & 1 & \cdots & 1 \\ 1 & \omega^{-1 \cdot 1} & \omega^{-1 \cdot 2} & \cdots & \omega^{-1 \cdot (N-1)} \\ 1 & \omega^{-2 \cdot 1} & \omega^{-2 \cdot 2} & \cdots & \omega^{-2 \cdot (N-1)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & \omega^{-(N-1) \cdot 1} & \omega^{-(N-1) \cdot 2} & \cdots & \omega^{-(N-1) \cdot (N-1)} \end{pmatrix} = \begin{pmatrix} f[0] \\ f[1] \\ f[2] \\ \vdots \\ f[N-1] \end{pmatrix}.$$

Por lo tanto, podemos escribir a las transformada de Fourier discreta como

$$\mathcal{F} = \begin{pmatrix} 1 & 1 & 1 & \cdots & 1 \\ 1 & \omega^{-1 \cdot 1} & \omega^{-1 \cdot 2} & \cdots & \omega^{-1 \cdot (N-1)} \\ 1 & \omega^{-2 \cdot 1} & \omega^{-2 \cdot 2} & \cdots & \omega^{-2 \cdot (N-1)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & \omega^{-(N-1) \cdot 1} & \omega^{-(N-1) \cdot 2} & \cdots & \omega^{-(N-1) \cdot (N-1)} \end{pmatrix}.$$

Entonces los elementos de matriz de \mathcal{F} son

$$\mathcal{F}_{mn} = \omega^{-mn}.$$

4.4.2. Transformada rápida de Fourier en una dimensión

Ahora sí, tenemos todas las herramientas para describir a las transformada rápida de Fourier. Veamos primero cómo reducir entonces la cantidad de cálculos que debemos realizar. Para poder comprender esto, es mejor centrarnos en un ejemplo. Consideraremos el caso absurdo en el que $N = 4$, entonces la matriz de la transformada discreta de Fourier es

$$\mathcal{F} = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & \omega^{-1 \cdot 1} & \omega^{-1 \cdot 2} & \omega^{-1 \cdot 3} \\ 1 & \omega^{-2 \cdot 1} & \omega^{-2 \cdot 2} & \omega^{-2 \cdot 3} \\ 1 & \omega^{-3 \cdot 1} & \omega^{-3 \cdot 2} & \omega^{-3 \cdot 3} \end{pmatrix} = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & \omega^{-1} & \omega^{-2} & \omega^{-3} \\ 1 & \omega^{-2} & \omega^{-4} & \omega^{-6} \\ 1 & \omega^{-3} & \omega^{-6} & \omega^{-9} \end{pmatrix},$$

4. Aplicación: tratamiento de imágenes

recordemos entonces que $\omega = e^{\frac{2\pi i}{N}} = e^{\frac{2\pi i}{4}}$, y que por lo tanto, $\omega^{-4} = 1$ de manera que

$$\omega^{-6} = \omega^{-4-2} = \omega^{-4}\omega^{-2} = \omega^{-2} \text{ y } \omega^{-9} = \omega^{-2 \cdot 4 - 1} = \omega^{-1}.$$

Entonces, en general, podemos reducir a módulo N los exponentes de las matrices, y con esta reducción la matriz es

$$\mathcal{F} = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & \omega^{-1} & \omega^{-2} & \omega^{-3} \\ 1 & \omega^{-2} & 1 & \omega^{-2} \\ 1 & \omega^{-3} & \omega^{-2} & \omega^{-1} \end{pmatrix},$$

pero también

$$\omega^{-2} = \left(e^{\frac{2\pi i}{4}}\right)^{-2} = e^{-\pi i} = 1,$$

lo que da finalmente

$$\mathcal{F} = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & \omega^{-1} & -1 & \omega^{-1} \\ 1 & -1 & 1 & -1 \\ 1 & \omega^{-1} & -1 & \omega^{-1} \end{pmatrix}.$$

Entonces hemos reducido la cantidad de cálculo de exponenciales complejas bastante. Sin embargo, hay mucho más que se puede hacer. Podemos observar que esta matriz tiene una estructura evidente. Vamos a aprovecharnos de esto. El truco es entonces reordenar a la transformada discreta de Fourier de orden N a una de orden $\frac{N}{2}$.

Antes de ver esto, vamos a necesitar de algunos conceptos algebraicos que utilizamos inconscientemente en el cálculo del ejemplo. Notemos que

$$\omega_{N/2}^{-1} = e^{-\frac{2\pi i}{N/2}} = e^{-\frac{4\pi i}{N}} = \omega_N^{-2}.$$

Entonces las potencias de $\omega_{N/2}^{-1}$ son potencias pares de ω_N^{-1} , de manera que si generalizamos

$$\omega_{N/2}^{-n} = \omega_N^{-2n} \text{ o bien } \omega_{N/2}^{-nm} = \omega_N^{-2nm}.$$

Ahora, veamos qué sucede con las potencias impares de ω_N^{-1} , $\omega_N^{-(2n+1)}$, entonces

$$\omega_N^{-(2n+1)} = \omega_N^{-1} \omega_N^{-2n},$$

por lo tanto

$$\omega_N^{-(2n+1)m} = \omega_N^{-m} \omega_{N/2}^{-nm}.$$

Finalmente, tenemos que

$$\omega_N^{N/2} = e^{\left(-\frac{2\pi i}{N}\right)\left(\frac{N}{2}\right)} = e^{-\pi i} = -1.$$

4.5. Convolución en versión discreta

Tenemos claramente dos casos para ω_N^m , uno para n par y otro para m impar. Así, podemos ver la transformada de Fourier $F[m]$ de la siguiente forma

$$F[m] = \sum_{n=0}^{N-1} f[n] \omega_N^{-nm} = \sum_{n=0}^{\frac{N}{2}-1} f[2n] \omega_N^{-(2n)m} + \sum_{n=0}^{\frac{N}{2}-1} f[2n] \omega_N^{-(2n+1)m}.$$

Hemos separado entonces a la transformada en dos sumas, una con índices pares y otros con índices impares. Ahora utilicemos las identidades que calculamos

$$\begin{aligned} F[m] &= \sum_{n=0}^{\frac{N}{2}-1} f[2n] \omega_N^{-(2n)m} + \sum_{n=0}^{\frac{N}{2}-1} f[2n] \omega_N^{-(2n+1)m} \\ F[m] &= \sum_{n=0}^{\frac{N}{2}-1} f[2n] \omega_{N/2}^{-nm} + \omega_N^{-m} \sum_{n=0}^{\frac{N}{2}-1} f[2n+1] \omega_{N/2}^{-nm}. \end{aligned}$$

Veamos con mayor detalle a estas dos sumas. Hay $N/2$ índices pares y $N/2$ índices impares, y separamos el arreglo N -dimensional a dos $N/2$ -dimensional. Entonces se reduce el orden de la matriz, y por lo tanto la cantidad de multiplicaciones que debe hacer la computadora.

4.4.3. Transformada rápida de Fourier en dos dimensiones

Hemos optimizado el tiempo de cómputo para calcular las transformadas de Fourier en una dimensión. Sin embargo, nuestras imágenes son de dos dimensiones. Entonces ¿cómo se implementa esto en dos dimensiones? Necesitamos la transformada de Fourier en una dimensión. Entonces, si $f(m, n)$ es una imagen de $M \times N$ entonces, tomamos la transformada de Fourier de cada una de las M filas \mathcal{F}_M y luego la transformada de Fourier de cada una de las N columnas \mathcal{F}_N . Entonces la transformada de Fourier en dos dimensiones es

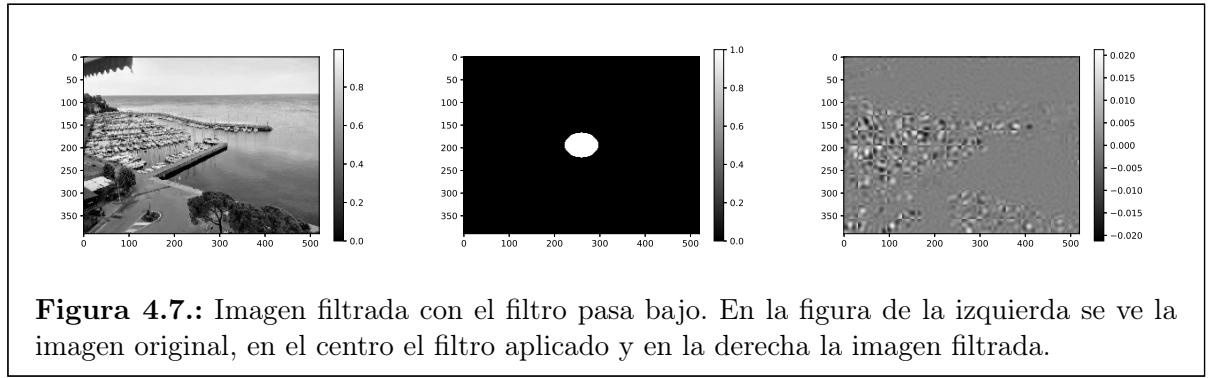
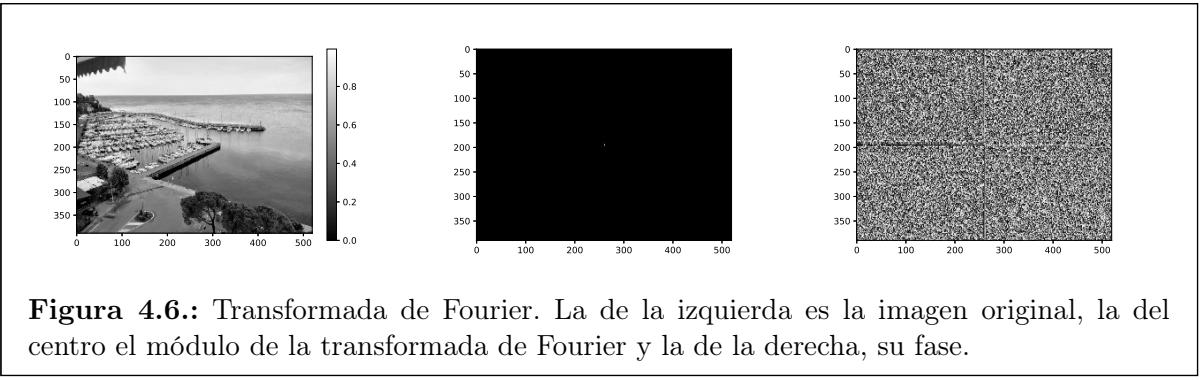
$$\mathcal{F}_M \cdot f \cdot \mathcal{F}_N.$$

Así es como implementamos rápidamente la transformada de Fourier en dos dimensiones, pero ¿para qué inventar el agua azucarada? Existen ya librerías muy eficientes para realizar la transformada de Fourier de una imagen. Entonces utilizaremos la librería `scipy` y dentro de esta, utilizamos `fftpack`. La transformada de Fourier de nuestra imagen predilecta se ve en la figura 4.6. Dado que el resultado es un arreglo de números complejos, para ilustrar la transformada de Fourier en dos dimensiones, debemos mostrar el módulo y la fase del resultado. El código puede verse en el apéndice A.4.5.

4.5. Convolución en versión discreta

Para realizar modificaciones en una imagen más complicados, se utiliza la convolución. Una convolución discreta en dos dimensiones se define de la siguiente manera.

4. Aplicación: tratamiento de imágenes



Definición 4.6. La convolución discreta en dos dimensiones es una imagen y se define como

$$f(x, y) = (g * h)(x, y) = \sum_m \sum_n g(x - m, y - n) h(m, n).$$

La convolución discreta cumple con las mismas propiedades que el caso continuo, particularmente con el teorema de convolución. Así que utilizaremos las transformadas de Fourier discretas para realizar los filtros. Veremos un solo ejemplo: el filtro pasa bajo. Vemos el efecto de este filtro en la figura 4.7. El código puede verse entonces en el apéndice A.4.6.

Bibliografía

- [Hsu et al.(1973)] Hsu, Mehra, and Torres] Hsu H, Mehra R, Torres R. Colección fondo e teoría y problemas con solución: Análisis de fourier[M/OL]. Fondo Educativo Interamericano, 1973. <https://books.google.com.gt/books?id=UeiHQgAACAAJ>
- [Osgood()] Osgood B. The fourier transform and its applications[J/OL]. EE261 - The Fourier Transform and its Applications. <https://see.stanford.edu/Course/EE261>.
- [Evans()] Evans T. Quantum field theory fourier transforms, delta functions and theta functions[EB/OL]. Imperial College London. <https://www.imperial.ac.uk/media/imperial-college/research-centres-and-groups/theoretical-physics/msc/current/qft/handouts/qftdeltafunction.pdf>.
- [Bueler()] Bueler E. The fourier transform of the heaviside function: a tragedy[EB/OL]. <https://www.cs.uaf.edu/~bueler/M611heaviside.pdf>.
- [Kreyszig(1971a)] Kreyszig E. Matemáticas avanzadas para ingeniería: Volumen i[M]. [S.l.]: Limusa-Wiley, 1971a
- [Kreyszig(1971b)] Kreyszig E. Matemáticas avanzadas para ingeniería: Volumen ii[M]. [S.l.]: Limusa-Wiley, 1971b
- [Mazet(2020)] Mazet V. Traitements et outils de base[EB/OL]. 2020. <http://miv.u-strasbg.fr/mazet/ofti/bases.pdf>.
- [Demmel()] Demmel J. Fast fourier transform of fft[EB/OL]. https://people.eecs.berkeley.edu/~demmel/cs170_spr07/LectureNotes/Lecture_FFT.pdf.

A. Códigos para gráficas

Las gráficas producidas para este documento son elaboradas en Python versión 3.6. Presentamos aquí los códigos para que el lector pueda replicarlas.

A.1. Capítulo 1

A.1.1. Figura 1.1

```
1 #!/usr/bin/env python3
2 # -*- coding: utf-8 -*-
3 """
4 Created on Sun Dec 15 20:53:09 2019
5
6 @author: juan
7 """
8
9 import numpy as np
10 import matplotlib.pyplot as plt
11
12 x1 = []
13 x2 = []
14
15 N = 1000
16 t = np.linspace(0,10,N)
17
18 for n in range(N):
19     x1.append(np.cos(2.0*np.pi*t[n]))
20     x2.append(0.5*np.cos(4.0*np.pi*t[n]))
21
22 SUM = np.array(x1) + np.array(x2)
23
24 fig, ax = plt.subplots(3,sharex=True,sharey=True, gridspec_kw={
25     'hspace':0})
26 ax[0].plot(t,x1)
27 ax[0].grid()
28 ax[1].plot(t,x2)
29 ax[1].grid()
30 ax[2].plot(t,SUM)
31 ax[2].grid()
```

A. Códigos para gráficas

```
31 plt.savefig('SuperposicionDeOndas.eps')
32 plt.show()
```

A.1.2. Figuras 1.2 y 1.3

```

1 #!/usr/bin/env python3
2 # -*- coding: utf-8 -*-
3 """
4 Created on Mon Dec 16 17:28:59 2019
5
6 @author: juan
7 """
8
9 import numpy as np
10 import matplotlib.pyplot as plt
11
12 x = []
13 x1 = []
14 x2 = []
15 x3 = []
16 x4 = []
17
18 N = 1000
19 w = 2.0*np.pi/2
20
21 t = np.linspace(0,4.0,N)
22
23 for i in range(len(t)):
24     x.append(0)
25     for n in range(2):
26         j = 2.0*n+1.0
27         x[i] += 4/np.pi * (1.0/j * np.sin(j*w*t[i]))
28
29 for i in range(len(t)):
30     x1.append(0)
31     for n in range(3):
32         j = 2.0*n+1.0
33         x1[i] += 4/np.pi * (1.0/j * np.sin(j*w*t[i]))
34
35 for i in range(len(t)):
36     x2.append(0)
37     for n in range(4):
38         j = 2.0*n+1.0
39         x2[i] += 4/np.pi * (1.0/j * np.sin(j*w*t[i]))
40
41 for i in range(len(t)):
42     x3.append(0)
43     for n in range(5):
44         j = 2.0*n+1.0
45         x3[i] += 4/np.pi * (1.0/j * np.sin(j*w*t[i]))
```

A. Códigos para gráficas

```
46
47 for i in range(len(t)):
48     x4.append(0)
49     for n in range(100):
50         j = 2.0*n+1.0
51         x4[i] += 4/np.pi * (1.0/j * np.sin(j*w*t[i]))
52
53
54 fig, ax = plt.subplots(5,sharex=True,sharey=True, gridspec_kw={
55     'hspace':0})
56 ax[0].plot(t,x)
57 ax[0].grid()
58 ax[1].plot(t,x1)
59 ax[1].grid()
60 ax[2].plot(t,x2)
61 ax[2].grid()
62 ax[3].plot(t,x3)
63 ax[3].grid()
64 ax[4].plot(t,x4)
65 ax[4].grid()
66 plt.savefig('Ejemplo1.eps')
67 plt.show()
68
69 x5 = []
70 M = int(N/4)
71
72 for n in range(M):
73     x5.append(1)
74 for n in range(M):
75     x5.append(-1)
76 for n in range(M):
77     x5.append(1)
78 for n in range(M):
79     x5.append(-1)
80
81 plt.plot(t,x5)
82 plt.grid()
83 plt.savefig('Ejemplo1Original.eps')
84 plt.show()
```

A.1.3. Figuras 1.4

```

1 #!/usr/bin/env python3
2 # -*- coding: utf-8 -*-
3 """
4 Created on Mon Dec 16 18:33:50 2019
5
6 @author: juan
7 """
8
9 import numpy as np
10 import matplotlib.pyplot as plt
11
12 x1 = []
13 x2 = []
14
15 N = 10000
16 w1 = 1
17 w2 = np.sqrt(2.0)
18 t = np.linspace(0,10,N)
19
20 for n in range(N):
21     x1.append(np.cos(w1*t[n]))
22     x2.append(np.cos(w2*np.pi*t[n]))
23
24 SUM = np.array(x1) + np.array(x2)
25
26 fig, ax = plt.subplots(3,sharex=True,sharey=True, gridspec_kw={
27     'hspace':0})
28 ax[0].plot(t,x1)
29 ax[0].grid()
30 ax[1].plot(t,x2)
31 ax[1].grid()
32 ax[2].plot(t,SUM)
33 ax[2].grid()
34 plt.savefig('NoPeriodica.eps')
35 plt.show()
36
37 x1 = []
38 x2 = []
39 w1 = 2*np.sqrt(2.0)
40 w2 = np.sqrt(2.0)
41
42 for n in range(N):
43     x1.append(np.cos(w1*t[n]))
44     x2.append(np.cos(w2*t[n]))

```

A. Códigos para gráficas

```
45
46 SUM = np.array(x1) + np.array(x2)
47
48 fig, ax = plt.subplots(3, sharex=True, sharey=True, gridspec_kw={'  
    hspace':0})
49 ax[0].plot(t,x1)
50 ax[0].grid()
51 ax[1].plot(t,x2)
52 ax[1].grid()
53 ax[2].plot(t,SUM)
54 ax[2].grid()
55 plt.savefig('Periodica.eps')
56 plt.show()
```

A.1.4. Figuras 1.5 y 1.6

```

1 #!/usr/bin/env python3
2 # -*- coding: utf-8 -*-
3 """
4 Created on Wed Dec 18 16:44:36 2019
5
6 @author: juan
7 """
8
9 import numpy as np
10 import matplotlib.pyplot as plt
11
12 N = 1000
13 t = np.linspace(-2.5,2.5,N)
14
15 f = []
16
17 for i in range(N):
18     f.append(t[i]-t[i]//1)
19
20 plt.plot(t,f)
21 plt.grid()
22 plt.savefig('FuncionSierra.eps')
23 plt.show()
24
25 M = [2,8,100]
26 fh = np.zeros((len(M),N))
27
28 for k in range(len(M)):
29     for i in range(N):
30         fh[k,i] = 0.5
31         for j in range(M[k]):
32             if j > 0:
33                 fh[k,i] += np.real(1.0j/(2*np.pi*j)*np.exp(2*np.pi*1.0j*j*t[i])) - 1.0j/(2*np.pi*j)*np.exp(-2*np.pi*1.0j*j*t[i]))
34
35 fig, ax = plt.subplots(4,sharex=True,sharey=True, gridspec_kw={
36     'hspace':0})
37 ax[0].plot(t,f)
38 ax[0].grid()
39 ax[1].plot(t,fh[0])
40 ax[1].grid()
41 ax[2].plot(t,fh[1])
42 ax[2].grid()
43 ax[3].plot(t,fh[2])
44 ax[3].grid()

```

A. Códigos para gráficas

```
44 plt.savefig('Ejemplo2.eps')
45 plt.show()
```

A.1.5. Figura 1.7

```
1 #!/usr/bin/env python3
2 # -*- coding: utf-8 -*-
3 """
4 Created on Wed Dec 18 12:04:51 2019
5
6 @author: juan
7 """
8
9 import numpy as np
10 import matplotlib.pyplot as plt
11
12 N = 20
13
14 c = []
15 n = []
16
17 for i in range(2*N+1):
18     n.append(i-N)
19     if n[i] % 2 == 0:
20         c.append(0.0)
21     else:
22         c.append((4/(n[i]*np.pi))**2)
23
24 fig,ax = plt.subplots()
25 ax.stem(n,c)
26 plt.title('Espectro')
27 plt.grid()
28 plt.savefig('Espectro.eps')
29 plt.show()
```

A.2. Capítulo 2

A.2.1. Figuras 2.1, 2.2, 2.3 y 2.4

```
1 #!/usr/bin/env python3
2 # -*- coding: utf-8 -*-
3 """
4 Created on Wed Dec 18 17:40:28 2019
5
6 @author: juan
7 """
8
9 import numpy as np
10 import matplotlib.pyplot as plt
11
12 N = 1000
13
14 t = np.linspace(-1.0,1.0,N)
15 f = []
16
17 for i in range(N):
18     if np.abs(t[i]) < 0.5:
19         f.append(1.0)
20     else:
21         f.append(0)
22
23 plt.plot(t,f)
24 plt.grid()
25 plt.savefig('FuncionRectangulo.eps')
26 plt.show()
27
28 fp = []
29 tp = np.linspace(-16.0,16.0,N)
30
31 for i in range(N):
32     if np.abs(tp[i]) < 0.5:
33         fp.append(1.0)
34     elif tp[i] < -14.5 and tp[i] > -15.5:
35         fp.append(1.0)
36     elif tp[i] > 14.5 and tp[i] < 15.5:
37         fp.append(1.0)
38     else:
39         fp.append(0.0)
40
41 plt.plot(tp,fp)
42 plt.grid()
43 plt.savefig('Periodizacion.eps')
```

```

44 plt.show()
45
46 M = 80
47
48 n = []
49 T = [2.0, 4.0, 8.0, 16.0]
50 c = np.zeros((len(T), 2*M+1))
51 w = np.zeros((len(T), 2*M+1))
52
53 for k in range(len(T)):
54     n = []
55     for i in range(2*M+1):
56         n.append(i-M)
57         w[k,i] = 1.0/T[k] * n[i]
58         if n[i] != 0:
59             c[k,i] = T[k]/(n[i]*np.pi) * np.sin(n[i]*np.pi/T[k])
60         else:
61             c[k,i] = 1.0
62
63
64
65 fig, ax = plt.subplots(4, sharex=True, sharey=False, gridspec_kw={
66     'hspace':0})
66 ax[0].stem(w[0,70:90],c[0,70:90])
67 ax[0].grid()
68 ax[1].stem(w[1,60:100],c[1,60:100])
69 ax[1].grid()
70 ax[2].stem(w[2,40:120],c[2,40:120])
71 ax[2].grid()
72 ax[3].stem(w[3],c[3])
73 ax[3].grid()
74 plt.xlabel('Frecuencia')
75 plt.ylabel('Coeficientes normalizados')
76 plt.savefig('Periodos.eps')
77 plt.show()
78
79 sinc = []
80 tsinc = [5*i for i in t]
81
82 for i in range(N):
83     sinc.append(np.sin(np.pi*tsinc[i])/(np.pi*tsinc[i]))
84
85 plt.plot(tsinc,sinc)
86 plt.ylabel('sinc($x$)')
87 plt.xlabel('$x$')
88 plt.grid()
89 plt.savefig('sinc.eps')

```

A. Códigos para gráficas

90 plt.show()

A.2.2. Figuras 2.5 y 2.6

```

1 #!/usr/bin/env python3
2 # -*- coding: utf-8 -*-
3 """
4 Created on Sun Dec 29 00:40:58 2019
5
6 @author: juan
7 """
8
9 import numpy as np
10 import matplotlib.pyplot as plt
11
12 N = 10000
13 t = np.linspace(-1.0,1.0,N)
14
15 f = []
16
17 for i in range(N):
18     if np.abs(t[i]) < 1:
19         f.append(1.0 - np.abs(t[i]))
20     else:
21         f.append(0.0)
22
23 plt.plot(t,f)
24 plt.grid()
25 plt.savefig('FuncionTriangulo.eps')
26 plt.show()
27
28 sinc2 = []
29 tsinc2 = np.linspace(-5.0,5.0,N)
30
31 for i in range(N):
32     sinc2.append((np.sin(np.pi*tsinc2[i])/(np.pi*tsinc2[i]))**2)
33
34 plt.plot(tsinc2,sinc2)
35 plt.ylabel('$sinc^2(x)$')
36 plt.xlabel('$x$')
37 plt.grid()
38 plt.savefig('sinc2.eps')
39 plt.show()

```

A. Códigos para gráficas

A.2.3. Figuras 2.7 y 2.8

```
1 #!/usr/bin/env python3
2 # -*- coding: utf-8 -*-
3 """
4 Created on Sun Dec 29 13:20:13 2019
5
6 @author: juan
7 """
8
9 import numpy as np
10 import matplotlib.pyplot as plt
11
12 N = 10000
13 t = np.linspace(-2.0,6.0,N)
14
15 a = [2.0,1.5,1.0,0.5,0.25]
16 f = np.zeros((len(a),N))
17
18 for A in range(len(a)):
19     for i in range(N):
20         if t[i] <= 0:
21             f[A,i] = 0.0
22         else:
23             f[A,i] = np.exp(-a[A]*t[i])
24
25 for i in range(len(a)):
26     plt.plot(t,f[i],label='$a$='+ str(a[i]))
27 plt.legend()
28 plt.grid()
29 plt.savefig('ExponencialDecay.eps')
30 plt.show()
31
32 L = np.zeros((len(a),N))
33 tL = np.linspace(-1.0,1.0,N)
34
35 for A in range(len(a)):
36     for i in range(N):
37         L[A,i] = 1.0/(a[A]**2 + 4*np.pi**2*tL[i]**2)
38
39 for i in range(len(a)):
40     plt.plot(tL,L[i],label='$a$='+ str(a[i]))
41 plt.legend()
42 plt.grid()
43 plt.savefig('Lorenziana.eps')
44 plt.show()
```

A.2.4. Figura 2.9

```
1 #!/usr/bin/env python3
2 # -*- coding: utf-8 -*-
3 """
4 Created on Sun Dec 29 13:56:36 2019
5
6 @author: juan
7 """
8
9 import numpy as np
10 import matplotlib.pyplot as plt
11
12 N = 10000
13 t = np.linspace(-3.0,3.0,N)
14 f = []
15
16 for i in range(N):
17     f.append(np.exp(-t[i]**2))
18
19 plt.plot(t,f)
20 plt.grid()
21 plt.savefig('DistribucionGaussiana.eps')
22 plt.show()
```

A. Códigos para gráficas

A.2.5. Figura 2.10

```
1 #!/usr/bin/env python3
2 # -*- coding: utf-8 -*-
3 """
4 Created on Sun Dec 29 17:58:24 2019
5
6 @author: juan
7 """
8
9 import numpy as np
10 import matplotlib.pyplot as plt
11
12 t = []
13 N = 10000
14
15 for i in range(N):
16     t.append(6.0*i/N-3.0) #no usamos linspace, porque sino no aparece
17         el 0
18 f = []
19
20 for i in range(N):
21     if t[i] == 0.0:
22         f.append(1000.0)
23     else:
24         f.append(0.0)
25
26 plt.plot(t,f)
27 plt.grid()
28 plt.savefig('deltaDirac.eps')
29 plt.show()
```

A.2.6. Figuras 2.11 y 2.12

```
1 #!/usr/bin/env python3
2 # -*- coding: utf-8 -*-
3 """
4 Created on Mon Dec 30 01:10:30 2019
5
6 @author: juan
7 """
8
9 import numpy as np
10 import matplotlib.pyplot as plt
11
12 N = 10000
13
14 t = np.linspace(-0.25,0.25,N)
15 f = []
16
17 for i in range(N):
18     if t[i] < 0:
19         f.append(0.0)
20     else:
21         f.append(1.0)
22
23 plt.plot(t,f)
24 plt.grid()
25 plt.savefig('Heaviside.eps')
26 plt.show()
27
28 FH = []
29
30 for i in range(N):
31     if t[i] == 0:
32         FH.append(0.0)
33     else:
34         FH.append(1/(2*np.pi*t[i]))
35
36 plt.plot(t,FH)
37 plt.grid()
38 plt.savefig('FTHeaviside.eps')
39 plt.show()
```

A. Códigos para gráficas

A.2.7. Figuras 2.13 y 2.14

```
1 e#!/usr/bin/env python3
2 # -*- coding: utf-8 -*-
3 """
4 Created on Sat Jan  4 02:27:13 2020
5
6 @author: juan
7 """
8
9 from mpl_toolkits.mplot3d import Axes3D
10 import matplotlib.pyplot as plt
11 from matplotlib import cm
12 from matplotlib.ticker import LinearLocator, FormatStrFormatter
13 import numpy as np
14
15
16 fig = plt.figure()
17 ax = fig.gca(projection='3d')
18
19 X = np.arange(-1.0, 1.0, 0.01)
20 Y = np.arange(-1.0, 1.0, 0.01)
21 Z = np.zeros((len(X), len(Y)))
22 for i in range(len(X)):
23     for j in range(len(Y)):
24         if np.abs(X[i])<0.5 and np.abs(Y[j])<0.5:
25             Z[i,j] = 1.0
26         else:
27             Z[i,j] = 0.0
28 X, Y = np.meshgrid(X, Y)
29
30 surf = ax.plot_surface(X, Y, Z, cmap='viridis', rstride=1, cstride
   =1,
                           linewidth=1, antialiased=False)
31 ax.set_zlim(-0.01, 1.01)
32 ax.xaxis.set_major_locator(LinearLocator(10))
33 ax.xaxis.set_major_formatter(FormatStrFormatter('%.02f'))
34 fig.colorbar(surf, shrink=0.5, aspect=5)
35 plt.savefig('Rect3D.eps')
36 plt.show()
37
38 fig = plt.figure()
39 ax = fig.gca(projection='3d')
40
41 X = np.arange(-5.0, 5.0, 0.1)
42 Y = np.arange(-5.0, 5.0, 0.1)
43 Z = np.zeros((len(X), len(Y)))
```

```
45 for i in range(len(X)):
46     for j in range(len(Y)):
47         Z[i,j] = np.sinc(X[i])*np.sinc(Y[j])
48 X, Y = np.meshgrid(X, Y)
49
50 surf = ax.plot_surface(X, Y, Z, cmap='viridis', rstride=1, cstride
51                         =1,
52                         linewidth=1, antialiased=False)
53 ax.set_zlim(-0.01, 1.01)
54 ax.xaxis.set_major_locator(LinearLocator(10))
55 ax.xaxis.set_major_formatter(FormatStrFormatter('%.02f'))
56 fig.colorbar(surf, shrink=0.5, aspect=5)
57 plt.savefig('sinc3D.eps')
58 plt.show()
```

A. Códigos para gráficas

A.2.8. Figura 2.15

```
1 #!/usr/bin/env python3
2 # -*- coding: utf-8 -*-
3 """
4 Created on Sat Jan  4 02:53:43 2020
5
6 @author: juan
7 """
8
9 from mpl_toolkits.mplot3d import Axes3D
10 import matplotlib.pyplot as plt
11 from matplotlib import cm
12 from matplotlib.ticker import LinearLocator, FormatStrFormatter
13 import numpy as np
14
15
16 fig = plt.figure()
17 ax = fig.gca(projection='3d')
18
19 a = 1.5
20
21 X = np.arange(-a, a, 0.1)
22 Y = np.arange(-a, a, 0.1)
23 X, Y = np.meshgrid(X, Y)
24 R = X**2 + Y**2
25 Z = np.exp(-np.pi*R)
26
27 surf = ax.plot_surface(X, Y, Z, cmap='viridis', rstride=1, cstride
28                         =1,
29                         linewidth=1, antialiased=False)
30 ax.set_zlim(-0.01, 1.01)
31 ax.xaxis.set_major_locator(LinearLocator(10))
32 ax.xaxis.set_major_formatter(FormatStrFormatter('%.02f'))
33 fig.colorbar(surf, shrink=0.5, aspect=5)
34 plt.savefig('Gauss3D.eps')
35 plt.show()
```

A.3. Capítulo 3

A.3.1. Figura 3.1

```

1 #!/usr/bin/env python3
2 # -*- coding: utf-8 -*-
3 """
4 Created on Tue Dec 31 21:53:09 2019
5
6 @author: juan
7 """
8
9 import numpy as np
10 import matplotlib.pyplot as plt
11
12 t = []
13 N = 1000
14
15 for i in range(N):
16     t.append(10.0*i/N-5.0)
17
18 f = []
19 ft = []
20 x = [0.5 for i in range(N)]
21 tt = [t[i]-x[i] for i in range(N)]
22 sigma = 0.1
23
24 for i in range(N):
25     f.append(t[i]-t[i]//1)
26     ft.append(tt[i]-tt[i]//1)
27
28 fr = []
29
30 for i in range(N):
31     fr.append((-tt[i])+tt[i]//1+1)
32
33 sinc = []
34
35 for i in range(N):
36     sinc.append(np.sin(np.pi*t[i])/(np.pi*t[i]))
37
38 mult = [sinc[i]*fr[i] for i in range(N)]
39
40 fig, ax = plt.subplots(4, sharex=True, sharey=True, gridspec_kw={
41     'hspace':0})
42 ax[0].plot(t,f)
43 ax[0].grid()

```

A. Códigos para gráficas

```
43 ax[1].plot(t,ft)
44 ax[1].grid()
45 ax[2].plot(t,fr)
46 ax[2].grid()
47 ax[3].plot(t,mult)
48 ax[3].grid()
49 plt.savefig('FuncionMultiplicada.eps')
50 plt.show()
```

A.3.2. Figura 3.2

```

1 #!/usr/bin/env python3
2 # -*- coding: utf-8 -*-
3 """
4 Created on Thu Jan  2 01:11:49 2020
5
6 @author: juan
7 """
8
9 import numpy as np
10 import matplotlib.pyplot as plt
11
12 t = []
13 N = 1000
14
15 for i in range(N):
16     t.append(10.0*i/N-5.0)
17
18 f = []
19 ft = []
20 x = [0.5 for i in range(N)]
21 tt = [t[i]-x[i] for i in range(N)]
22 sigma = 0.1
23
24 for i in range(N):
25     f.append(1.0/(sigma*np.sqrt(2*np.pi)) * np.exp(-t[i]**2/(2*sigma
26             **2)))
27     ft.append(1.0/(sigma*np.sqrt(2*np.pi)) * np.exp(-tt[i]**2/(2*
28             sigma**2)))
29
30 fr = []
31
32 for i in range(N):
33     fr.append(1.0/(sigma*np.sqrt(2*np.pi)) * np.exp(-(-t[i]-0.5)
34             **2/(2*sigma**2)))
35
36 sinc = []
37
38 for i in range(N):
39     sinc.append(np.sin(np.pi*t[i])/(np.pi*t[i]))
40
41 mult = [sinc[i]*fr[i] for i in range(N)]
42
43 fig, ax = plt.subplots(4, sharex=True, sharey=True, gridspec_kw={
44     'hspace':0})
45 ax[0].plot(t,f)

```

A. Códigos para gráficas

```
42 ax[0].grid()
43 ax[1].plot(t,ft)
44 ax[1].grid()
45 ax[2].plot(t,fr)
46 ax[2].grid()
47 ax[3].plot(t,mult)
48 ax[3].grid()
49 plt.savefig('FuncionMultiplicadaGaussiana.eps')
50 plt.show()
```

A.3.3. Figuras 3.4 y 3.5

```

1 #!/usr/bin/env python3
2 # -*- coding: utf-8 -*-
3 """
4 Created on Fri Jan  3 21:26:19 2020
5
6 @author: juan
7 """
8
9 import numpy as np
10 import matplotlib.pyplot as plt
11
12 N = 1000
13
14 s = np.linspace(-15.0,15.0,N)
15 f = []
16 v0 = 10.0
17 vc = 2.0
18
19 for i in range(N):
20     if np.abs(s[i]) < v0+vc and np.abs(s[i])>v0-vc:
21         f.append(1.0)
22     else:
23         f.append(0)
24
25 plt.plot(s,f)
26 plt.grid()
27 plt.savefig('FuncionTransfPasaBanda.eps')
28 plt.show()
29
30 t = np.linspace(-1.0,1.0,N)
31 b = []
32
33 for i in range(N):
34     b.append(4*vc*np.cos(2*np.pi*v0*t[i])*np.sinc(2*vc*t[i]))
35
36 plt.plot(t,b)
37 plt.grid()
38 plt.savefig('FuncionRespImpulsoPasaBanda.eps')
39 plt.show()

```

A. Códigos para gráficas

A.3.4. Figuras 3.6 y 3.7

```
1 #!/usr/bin/env python3
2 # -*- coding: utf-8 -*-
3 """
4 Created on Fri Jan  3 22:10:45 2020
5
6 @author: juan
7 """
8
9 import numpy as np
10 import matplotlib.pyplot as plt
11
12 N = 1000
13
14 s = np.linspace(-5.0,5.0,N)
15 f = []
16 vc = 2.0
17
18 for i in range(N):
19     if np.abs(s[i]) < vc:
20         f.append(0.0)
21     else:
22         f.append(1.0)
23
24 plt.plot(s,f)
25 plt.grid()
26 plt.savefig('FuncionTransfPasaAlto.eps')
27 plt.show()
28
29 t = []
30
31 for i in range(N):
32     t.append(2.0*i/N-1.0)
33
34 b = []
35
36 for i in range(N):
37     if t[i] == 0:
38         b.append(3 - np.sinc(2*vc*t[i]))
39     else:
40         b.append(- np.sinc(2*vc*t[i]))
41
42 plt.plot(t,b)
43 plt.grid()
44 plt.savefig('FuncionRespImpulsoPasaAlto.eps')
45 plt.show()
```

A.4. Capítulo 4

A.4.1. Figura 4.2

```

1 #!/usr/bin/env python3
2 # -*- coding: utf-8 -*-
3 """
4 Created on Sun Jan  5 16:52:11 2020
5
6 @author: juan
7 """
8
9 import numpy as np
10 import matplotlib.pyplot as plt
11 import skimage.io as io
12
13 #Leer la imagen
14 f = io.imread('Dia.jpg')
15
16 #Desplegar los valores de los pixeles en la imagen
17 print(f)
18 #Ver dimensiones de la imagen
19 Nx = len(f)
20 Ny = len(f[0])
21
22 print('La imagen es de ' + str(Nx) + 'x' + str(Ny) + ' pixeles.')
23 #Convertir la imagen a escala de grises en un primer intento
24 fGris = np.zeros((Nx,Ny))
25 for i in range(Nx):
26     for j in range(Ny):
27         fgris = 0
28         for k in range(3):
29             fgris += (f[i][j][k])/3/255
30         fGris[i][j] = fgris
31
32 plt.imshow(fGris, cmap='gray')
33 plt.colorbar()
34 plt.savefig('DiaGris.eps')
35 plt.show()
36
37 #Una mejor forma
38 fGris2 = np.zeros((Nx,Ny))
39 for i in range(Nx):
40     for j in range(Ny):
41         fgris = (0.2125*f[i][j][0] + 0.7154*f[i][j][1] + 0.0721*f[i]
42                         ][j][2])/255
43         fGris2[i][j] = fgris

```

A. Códigos para gráficas

```
43
44
45
46 plt.imshow(fGris2, cmap='gray')
47 plt.colorbar()
48 plt.savefig('DiaGrisMejor.eps')
49 plt.show()
50
51 #Muestra la imagen original
52 plt.imshow(f)
53 plt.savefig('Dia.eps')
54 plt.show()
```

A.4.2. Figura 4.3

```

1 #!/usr/bin/env python3
2 # -*- coding: utf-8 -*-
3 """
4 Created on Sun Jan  5 23:50:15 2020
5
6 @author: juan
7 """
8
9 import numpy as np
10 import matplotlib.pyplot as plt
11 import skimage.io as io
12
13 #Leer la imagen
14 f = io.imread('Dia.jpg')
15
16 #Ver dimensiones de la imagen
17 Nx = len(f)
18 Ny = len(f[0])
19
20 #Convertir la imagen a blanco y negro
21 fBN = np.zeros((Nx,Ny))
22 for i in range(Nx):
23     for j in range(Ny):
24         fgris = 0
25         for k in range(3):
26             fgris += f[i][j][k]/3/255
27         if fgris < 0.5:
28             fBN[i,j] = 0
29         else:
30             fBN[i,j] = 1.0
31
32 plt.imshow(fBN, cmap='gray')
33 plt.colorbar()
34 plt.savefig('DiaBN.eps')
35 plt.show()

```

A. Códigos para gráficas

A.4.3. Figura 4.4

```
1 #!/usr/bin/env python3
2 # -*- coding: utf-8 -*-
3 """
4 Created on Mon Jan  6 00:09:36 2020
5
6 @author: juan
7 """
8
9 import numpy as np
10 import matplotlib.pyplot as plt
11 import skimage.io as io
12
13 #Leer la imagen
14 f = io.imread('Dia.jpg')
15
16 #Ver dimensiones de la imagen
17 Nx = len(f)
18 Ny = len(f[0])
19
20 #Convertir la imagen a negativo
21 fN = np.zeros((Nx,Ny,3))
22 for i in range(Nx):
23     for j in range(Ny):
24         fgris = 0
25         for k in range(3):
26             fN[i][j][k] = 1-f[i][j][k]/255
27
28 plt.imshow(fN)
29 plt.savefig('DiaNegativo.eps')
30 plt.show()
```

A.4.4. Figura 4.5

```

1 #!/usr/bin/env python3
2 # -*- coding: utf-8 -*-
3 """
4 Created on Mon Jan  6 00:25:43 2020
5
6 @author: juan
7 """
8
9 import numpy as np
10 import matplotlib.pyplot as plt
11 import skimage.io as io
12
13 #Leer la imagen
14 f = io.imread('Dia.jpg')
15
16 #Ver dimensiones de la imagen
17 Nx = len(f)
18 Ny = len(f[0])
19
20 #Convertimos la imagen a escala de grises
21 fGris = np.zeros((Nx,Ny))
22 for i in range(Nx):
23     for j in range(Ny):
24         fgris = (0.2125*f[i][j][0] + 0.7154*f[i][j][1] + 0.0721*f[i]
25             ][j][2])/255
26         fGris[i][j] = fgris
27
28
29 plt.imshow(fGris, cmap='gray')
30 plt.colorbar()
31 plt.savefig('DiaGrisMejor.eps')
32 plt.show()
33
34 #Construyamos el histograma
35 plt.hist(fGris,bins=256)
36 plt.savefig('HistogramaGris.eps')
37 plt.show()
38
39 #
40 #-----#
41 #NEGATIVO
42 #
43 #-----#

```

A. Códigos para gráficas

```
42 fN = np.zeros((Nx,Ny))
43 for i in range(Nx):
44     for j in range(Ny):
45         neg = 1-fGris[i][j]
46         fN[i][j] = neg
47
48 plt.imshow(fN,cmap='gray')
49 plt.colorbar()
50 plt.savefig('DiaGrisNegativo.eps')
51 plt.show()
52
53
54 #Construimos el histograma
55 plt.hist(fN,bins=256)
56 plt.savefig('HistogramaNegativo.eps')
57 plt.show()
58
59 #Funcion aplicada
60 x = np.arange(0.0, 1.0, 0.01)
61 y = 1-x
62 plt.plot(x,y)
63 plt.grid()
64 plt.savefig('FuncionNegativo.eps')
65 plt.show()
66
67 #
-----#
68 #CORRECCION GAMMA
69 #
-----#
70
71 gamma = 0.4
72 fGamma = np.zeros((Nx,Ny))
73
74 for i in range(Nx):
75     for j in range(Ny):
76         fg = fGris[i][j]**gamma
77         fGamma[i][j] = fg
78
79 plt.imshow(fGamma,cmap='gray')
80 plt.colorbar()
81 plt.savefig('DiaGrisGamma.eps')
82 plt.show()
83
```

```

84 #Construimos el histograma
85 plt.hist(fGamma,bins=256)
86 plt.savefig('HistogramaGamma.eps')
87 plt.show()
88
89 #Funcion aplicada
90 x = np.arange(0.0, 1.0, 0.01)
91 y = x**gamma
92 plt.plot(x,y)
93 plt.grid()
94 plt.savefig('FuncionGamma.eps')
95 plt.show()
96
97 #
-----#
98 #NORMALIZACION
99 #
-----#
100
101 MIN = fGris.min()
102 MAX = fGris.max()
103 fNorm = np.zeros((Nx,Ny))
104
105 for i in range(Nx):
106     for j in range(Ny):
107         norm = (fGris[i][j]-MIN)/(MAX-MIN)
108         fNorm[i][j] = norm
109
110 plt.imshow(fNorm,cmap='gray')
111 plt.colorbar()
112 plt.savefig('DiaGrisNorm.eps')
113 plt.show()
114
115 #Construimos el histograma
116 plt.hist(fNorm,bins=256)
117 plt.savefig('HistogramaNorm.eps')
118 plt.show()
119
120 #Funcion aplicada
121 x = np.arange(0.0, 1.0, 0.01)
122 y = (x-MIN)/(MAX - MIN)
123 plt.plot(x,y)
124 plt.grid()
125 plt.savefig('FuncionNorm.eps')
126 plt.show()

```

A. Códigos para gráficas

A.4.5. Figura 4.6

```
1 #!/usr/bin/env python3
2 # -*- coding: utf-8 -*-
3 """
4 Created on Mon Jan  6 17:14:30 2020
5
6 @author: juan
7 """
8
9 import numpy as np
10 import matplotlib.pyplot as plt
11 import skimage.io as io
12 from scipy import fftpack
13
14 #Leer la imagen
15 f = io.imread('Dia.jpg')
16
17 #Ver dimensiones de la imagen
18 Nx = len(f)
19 Ny = len(f[0])
20
21 #Convertimos la imagen a escala de grises
22 fGris = np.zeros((Nx,Ny))
23 for i in range(Nx):
24     for j in range(Ny):
25         fgris = (0.2125*f[i][j][0] + 0.7154*f[i][j][1] + 0.0721*f[i]
26                   ][j][2])/255
27         fGris[i][j] = fgris
28
29 plt.imshow(fGris, cmap='gray')
30 plt.colorbar()
31 plt.savefig('DiaGrisMejor.eps')
32 plt.show()
33
34 #Transformada rapida de Fourier
35 ftGris = fftpack.fft2(fGris)
36 FTGris = fftpack.fftshift(ftGris)
37
38 FTGrisMod = np.abs(FTGris)
39 FTGrisFase = np.angle(FTGris)
40
41 plt.imshow(FTGrisMod, cmap = 'gray')
42 plt.savefig('DiaGrisFTMod.eps')
43 plt.show()
44
```

```
45 plt.imshow(FTGrisFase, cmap = 'gray')
46 plt.savefig('DiaGrisFTFase.eps')
47 plt.show()
```

A. Códigos para gráficas

A.4.6. Figura 4.7

```
1 #!/usr/bin/env python3
2 # -*- coding: utf-8 -*-
3 """
4 Created on Wed Jan  8 01:32:30 2020
5
6 @author: juan
7 """
8
9 import numpy as np
10 import matplotlib.pyplot as plt
11 import skimage.io as io
12 from scipy import fftpack
13
14 #Leer la imagen
15 f = io.imread('Dia.jpg')
16
17 #Ver dimensiones de la imagen
18 Nx = len(f)
19 Ny = len(f[0])
20
21 #Convertimos la imagen a escala de grises
22 fGris = np.zeros((Nx,Ny))
23 for i in range(Nx):
24     for j in range(Ny):
25         fgris = (0.2125*f[i][j][0] + 0.7154*f[i][j][1] + 0.0721*f[i]
26                   ][j][2])/255
27         fGris[i][j] = fgris
28
29 plt.imshow(fGris, cmap='gray')
30 plt.colorbar()
31 plt.savefig('DiaGrisMejor.eps')
32 plt.show()
33
34 #Transformada rapida de Fourier
35 ftGris = fftpack.fft2(fGris)
36 FTGris = fftpack.fftshift(ftGris)
37
38 FTGrisMod = np.abs(FTGris)
39 FTGrisFase = np.angle(FTGris)
40
41 plt.imshow(FTGrisMod, cmap = 'gray')
42 plt.savefig('DiaGrisFTMod.eps')
43 plt.show()
44
```

```
45 plt.imshow(FTGrisFase, cmap = 'gray')
46 plt.savefig('DiaGrisFTFase.eps')
47 plt.show()
48
49 #Filtro
50
51 X = np.linspace(-1.0,1.0,Nx)
52 Y = np.linspace(-1.0,1.0,Ny)
53 Z = np.zeros((len(X),len(Y)))
54 for i in range(len(X)):
55     for j in range(len(Y)):
56         if np.abs(X[i]**2+Y[j]**2)<0.02:
57             Z[i,j] = 1.0
58         else:
59             Z[i,j] = 0.0
60
61 plt.imshow(Z, cmap='gray')
62 plt.colorbar()
63 plt.savefig('PasaBajo.eps')
64 plt.show()
65
66 #Filtrado
67
68 ImFilt = np.multiply(Z,ftGris)
69 imagen = fftpack.ifft2(ImFilt)
70 Imagen = np.real(imagen)
71
72 plt.imshow(Imagen, cmap='gray')
73 plt.colorbar()
74 plt.savefig('ImagenFiltrada.eps')
75 plt.show()
```


Índice alfabético

- asociatividad, 55
- base, 2
- base de Fourier, 4
- base ortonormal, 4
- coeficientes de Fourier, 5, 10
- condiciones de Dirichlet, 11
- comutatividad, 55
- convergencia, 11
- convolución, 50, 56
- convolución discreta, 74
- coordenadas polares, 43
- decaimiento exponencial, 30
- delta de Dirac, 35, 36, 55
- distributividad, 55
- ecuación de calor, 14, 15
- elemento neutro, 55
- escalón unitario, 37
- espectro, 15, 16, 20, 23, 25
- fase, 57
- fenómeno de Gibbs, 6, 10
- filtro, 56
- filtro elimina banda, 60
- filtro pasa alto, 59
- filtro pasa bajo, 57, 60
- filtro pasa banda, 58
- frecuencia, 7, 15, 23
- frecuencia angular, 3, 7
- frecuencia de corte, 57
- función de Bessel, 45
- función de respuesta de impulso, 60
- función de transferencia, 56, 59
- función gaussiana, 31, 33, 35, 42
- función impulso, 35
- función lorenziana, 31, 33
- función periódica, 3, 8
- función radial, 44
- función rectángulo, 19, 21, 41
- función sierra, 10
- función triángulo, 29, 30
- histograma, 66
- identidad de Parseval, 25
- imagen digital, 63
- ímpar, 26
- matriz de la transformada discreta de Fourier, 71
- par, 26
- paridad, 27
- período, 3, 15
- periodización, 19
- pixel, 63
- principio de superposición, 1
- producto escalar, 3
- Python, 77
- respuesta de impulso, 56
- rotación, 28
- señal de entrada, 56
- señal de salida, 56
- señal invertida, 26

Índice alfabético

separabilidad, 69
serie de Fourier, 20
serie de Fourier, 5, 7, 9, 21
serie de Fourier n-dimensional, 16

teorema de convolución, 50, 51, 61
teorema de inversión de Fourier., 25
teorema de similitud, 28, 40
teorema de traslación, 28
transformación, 67
transformación lineal, 27
transformada de Fourier, 19, 22, 23, 26–
28, 33, 35, 36, 38, 49, 51, 56
transformada de Fourier discreta, 68
transformada de Fourier en dos dimensio-
nes, 73
transformada discreta de Fourier, 69
transformada inversa, 26, 51
transformada inversa de Fourier, 23, 24,
27, 39
transformada inversa de Fourier discreta,
68
transformada rápida de Fourier, 69, 71
traslación, 28
tratamiento de imágenes, 63