

## Índice de <https://www.fceia.unr.edu.ar/acustica/psychoacoustics/>

<b>Tipo</b>	<b>Función, script o archivo</b>	<b>Descripción</b>
<b>Material de estudio</b>	<a href="#">Indice.pdf</a>	Este archivo
	<a href="#">Leerme.txt</a>	Instrucciones básicas para empezar a usar Scilab y las funciones y enlaces de este directorio
	<a href="#">Introduccion_a_Scilab_(Federico_Miyara).pdf</a>	Un texto introductorio sobre Scilab que cubre la mayoría del material necesario para utilizar el lenguaje y el entorno de programación
	<a href="#">Senales.ppt</a>	Presentación multimedial sobre señales en tiempo continuo, tiempo discreto y digitales
	<a href="#">LTI.ppt</a>	Presentación multimedial sobre sistemas lineales invariantes en el tiempo
	<a href="#">Espectro.ppt</a>	Presentación multimedial introductoria sobre espectro de señales continuas y discretas, periódicas y no periódicas
	<a href="#">alias_vs_noalias_no_spectral_leakage.sci</a>	Ejemplo de señales sin alias y con alias
<b>Manejo de archivos</b>	<a href="#">flacread.sci</a>	Lee un archivo de audio en formato FLAC. Requiere tener instalado el códec de referencia de FLAC
	<a href="#">flacwrite.sci</a>	Exporta una señal a un archivo FLAC. Requiere tener instalado el códec de referencia de FLAC
	<a href="#">flac2wav.sci</a>	Convierte un archivo FLAC en wav. Requiere tener instalado el códec de referencia de FLAC
	<a href="#">wav2flac.sci</a>	Convierte un archivo wav en FLAC. Requiere tener instalado el códec de referencia de FLAC
	<a href="#">cueread.sci</a>	Lee y tabula cues en un archivo de audio que los tenga.
	<a href="#">bmpread.sci</a>	Permite leer un bitmap en formato BMP
	<a href="#">bmp2table.sci</a>	Convierte una curva escaneada (y depurada) en una tabla de valores. Sirve para parametrizar curvas empíricas.
<b>Procesamiento de audio básico</b>	<a href="#">fade.sci</a>	Agrega fade-in y fade-out a una señal de audio
	<a href="#">trim_audio.sci</a>	Remueve porciones de una señal de audio por debajo de un umbral especificado si son suficientemente largas
	<a href="#">event_mix.sci</a>	Mezclador muy flexible de eventos sonoros de cualquier duración, cualquier instante inicial y cualquier ganancia. Resuelve el problema de sumar señales que no están perfectamente alineadas sin tener que agregar ceros al principio y al final. Sirve para crear paisajes sonoros
	<a href="#">mix.sci</a>	Mezclador flexible de dos señales

<b>Tipo</b>	<b>Función, script o archivo</b>	<b>Descripción</b>
<b>Reverberación</b>	<a href="#">wavmix.sci</a>	Mezclador flexible de dos señales de audio aunque tengan diferente longitud, empiecen en diferentes instantes y sean mono o estéreo.
	<a href="#">wavpan.sci</a>	Panea una señal mono a estéreo
	<a href="#">wavappend.sci</a>	Agrega una señal de audio a un archivo wav existente. Sirve para ir agregando segmentos de señal a un archivo wav muy grande sin tener que cargar todo el archivo en memoria
	<a href="#">psicoenhancer.sci</a>	Realizador psicoacústico de audio
	<a href="#">envelope.sci</a>	Determina la envolvente energética de una señal
<b>Señales</b>	<a href="#">late_reverb.sci</a>	Calcula la cola de la reverberación (reverberación tardía)
	<a href="#">late_reverb_prueba.sce</a>	Prueba de la función anterior
	<a href="#">coupled_reverb.sci</a>	Obtiene la reverberación correspondiente a dos recintos acoplados acústicamente por una abertura o un tabique liviano
	<a href="#">iso3382.sci</a>	Aplica el método de la ISO 3382 para la determinación de la reverberación en bandas de octava a partir de la respuesta al impulso
	<a href="#">iso3382_prueba.sce</a>	Prueba de ISO3382
	<a href="#">iso3382_test.sce</a>	Prueba de ISO 3382
	<a href="#">bonello.sci</a>	Permite aplicar el criterio de Bonello a salas pequeñas para determinar si el sonido es difuso o existen resonancias
<b>Señales</b>	<a href="#">signals.sce</a>	Ejemplos de señales
	<a href="#">bandnoise.sci</a>	Crea bandas de ruido de diferentes tipos: octava, fracción de octava, críticas
	<a href="#">pulses.sci</a>	Obtiene un tren de pulsos limitado en banda. Permite aplicarle a un filtro y obtener a la salida una señal periódica con espectro dado.
	<a href="#">whitenoise.sci</a>	Ruido blanco gaussiano
	<a href="#">whitenoise_perfect.sci</a>	Ruido blanco espectralmente perfecto en una sola ventana FFT
	<a href="#">mls.sci</a>	Obtiene secuencias de máxima longitud, un tipo de ruido discreto formado por unos y ceros distribuidos de una forma tal que el resultado tiene espectro constante con la frecuencia (blanco). La diferencia con el ruido blanco aleatorio es que su amplitud está acotada, mientras que en el ruido aleatorio teóricamente pueden aparecer valores mucho más altos que el valor eficaz.
	<a href="#">pinknoise.sci</a>	Ruido rosa (densidad espectral $K/f$ )
	<a href="#">browniannoise.sci</a>	Ruido browniano (densidad espectral $K/f^2$ )

<b>Tipo</b>	<b>Función, script o archivo</b>	<b>Descripción</b>
	<a href="#">velvetnoise.sci</a>	Obtiene ruido terciopelo, un tipo de ruido blanco “ralo”, formado por pulsos separados cuasi aleatoriamente. Se usa en la simulación de reverberación
	<a href="#">notched_noise.sci</a>	Ruido con un pozo espectral. Es útil para ciertos experimentos psicoacústicos
	<a href="#">notched_noise_prueba.sce</a>	Prueba de la función anterior
	<a href="#">band_noise_of_various_bandwidths.sce</a>	Ejemplos de bandas de ruido de diversos anchos
	<a href="#">pulsed_signal.sci</a>	Genera una señal pulsada (por ejemplo tonos de corta duración repetitivos, ruido blanco, ruido de banda)
	<a href="#">iec_60645_3.sce</a>	Obtiene un pulso tonal de 5 ciclos con fade-in y fade-out
	<a href="#">uen.sci</a>	Obtiene el ruido de excitación uniforme, un ruido cuyo espectro es tal que produce excitación uniforme en todas las frecuencias
	<a href="#">AM.sce</a>	Ejemplos de modulación de amplitud
	<a href="#">FM.sce</a>	Ejemplos de modulación de frecuencia
	<a href="#">fmod.sci</a>	Modulador de frecuencia
<b>Análisis de espectro</b>	<a href="#">fftmean.sci</a>	Calcula el promedio de la FFT de una señal más larga que la ventana de la FFT. Admite promedio aritmético (lineal) o energético, lo cual permite obtener densidad espectral
	<a href="#">fftmeandB.sci</a>	Calcula el promedio de la FFT de una señal más larga que la ventana de la FFT en dB
	<a href="#">octaves.sci</a>	Obtiene el espectro de bandas de octava de una sola ventana
	<a href="#">octavemean.sci</a>	Calcula el espectro de bandas de octava promedio de una señal
	<a href="#">spectralpeaks.sci</a>	Determina picos espectrales (candidatos a ser armónicos o parciales)
	<a href="#">spectralpeaks_prueba.sce</a>	Prueba de la función anterior
	<a href="#">spectrogram.sci</a>	Permite graficar el espectrograma de una señal, es decir el espectro en función del tiempo
	<a href="#">spectrogram_prueba.sce</a>	Prueba de la función spectrogram
	<a href="#">bands2spectdens.sci</a>	Permite convertir un espectro de bandas de octava a densidad espectral mediante una adecuada interpolación
	<a href="#">blackman.sci</a>	Obtiene la ventana de Blackman periódica
	<a href="#">hann.sci</a>	Obtiene la ventana de Hann periódica

Tipo	Función, script o archivo	Descripción
	<a href="#">lpc.sci</a>	Obtiene el modelo LPC (linear prediction code) de una señal. Es útil para estimar, a partir de la señal, el filtro que le da origen suponiendo que la fuente es un tren de pulsos para señales periódicas o un ruido blanco para señales de banda ancha
Filtros	<a href="#">fftfilter.sci</a>	Filtro FFT para señales de longitud arbitraria
	<a href="#">fftfilter_prueba.sci</a>	Prueba el filtro FFT
	<a href="#">bandfilter.sci</a>	Filtro de banda
	<a href="#">morphfilter.sci</a>	Filtro variable en el tiempo. La respuesta en frecuencia va variando a lo largo del tiempo según lo requerido
	<a href="#">morphfilter_prueba.sce</a>	Prueba el filtro anterior
	<a href="#">trackfilter.sci</a>	Filtra una señal siguiendo la evolución especificada de una frecuencia. Permite extraer armónicos o múltiples armónicos
	<a href="#">trackfilter_prueba.sce</a>	Prueba de la función anterior
	<a href="#">bands.sci</a>	Descompone una señal en bandas espectrales de distinto tipo
	<a href="#">Afilter.sci</a>	Aplica filtros A y C a una señal
	<a href="#">Aweighting.sci</a>	Ganancia del filtro A
	<a href="#">imp_resp.sci</a>	Calcula la respuesta al impulso a partir de la función de transferencia compleja de un sistema (en amplitud y fase)
	<a href="#">band_tone_resp.sci</a>	Calcula la respuesta a un tono analítica de un filtro de banda.
	<a href="#">band_imp_resp.sci</a>	Respuesta al impulso analítica de un filtro de banda. Combinada con la convolución (convol) permite obtener la respuesta de un filtro continuo analítico ante cualquier señal
	<a href="#">biquad_imp_resp.sci</a>	Respuesta al impulso analítica de un filtro genérico compuesto por células biquadráticas. Combinada con la convolución (convol) permite obtener la respuesta de un filtro analítico biquadrático ante cualquier señal
	<a href="#">filter_imp_resp.sci</a>	Respuesta al impulso analítica de un filtro genérico compuesto por polos de orden 1 (en algunos casos pueden ser complejos conjugados). Combinada con la convolución (convol) permite obtener la respuesta de un filtro analítico genérico ante cualquier señal
	<a href="#">filter_imp_resp_prueba.sci</a>	Prueba de la función anterior
	<a href="#">timevarconv.sci</a>	Convolución variable en el tiempo. Sirve para sistemas no invariantes en el tiempo, es decir
	<a href="#">varconv.sci</a>	cuya respuesta al impulso va evolucionando en el tiempo

Tipo	Función, script o archivo	Descripción
	<a href="#">timevarconv_test.sce</a> <a href="#">varconv_test.sce</a>	Prueba de la anterior
	<a href="#">conv_delay.sci</a>	Convolución con una respuesta al impulso retardada (o anticipada)
	<a href="#">TDH39_equalizer.sce</a>	Ecualizador para auriculares audiométricos TDH 39. Permite aplanar la respuesta.
	<a href="#">DT48_equalizer.sce</a>	Ecualizador para el auricular audiométrico DT48. Permite aplanar la respuesta
	<a href="#">resample_filter.sci</a>	Convierte los coeficientes z de un filtro discreto de una tasa de muestreo a otra.
	<a href="#">critenv.sci</a>	Descompone una señal en bandas críticas y obtiene sus envolventes
	<a href="#">spectral_blur.sci</a>	Borroneo espectral: reemplaza cada línea espectral por una banda de igual potencia de modo que el espectro queda desdibujado. Los armónicos son, así, reemplazados por bandas de ruido de ancho controlable
	<a href="#">spectral_blur.sce</a> <a href="#">spectral_blur_test.sce</a>	Ejemplos de borroneo espectral
	<a href="#">time_blur.sce</a>	Borroneo temporal: suaviza las envolventes de una señal. Prueba de concepto
	<a href="#">iir_filter_prueba.sce</a>	Demostración de uso de las funciones iir() y filter() de Scilab
<b>Psicoacústica</b>	<a href="#">iso226.sci</a>	Obtiene el nivel de sonoridad en función del nivel de presión sonora para cada frecuencia según la norma ISO 226 en cuatro versiones desde 1961 hasta 2023
	<a href="#">iso226.sce</a>	Obtiene las curvas isofónicas en cuatro versiones desde 1961 a 2023
	<a href="#">iso226_test.sci</a>	Prueba de iso226
	<a href="#">critband.sci</a>	Obtiene el ancho de banda de la banda crítica centrada en una frecuencia
	<a href="#">zbark.sci</a>	Convierte la frecuencia en variable psicoacústica z (índice de banda crítica o critical band rate)
	<a href="#">zkrab.sci</a>	Convierte la variable psicoacústica z (índice de banda crítica o critical band rate) en el valor correspondiente de frecuencia
	<a href="#">mask.sci</a>	Obtiene el patrón de enmascaramiento, es decir el umbral auditivo en presencia de uno o varios tonos máscara
	<a href="#">loudness.sci</a>	Calcula sonoridad en función del espectro en bandas de octava por el método de Stevens (ISO 532 <sup>a</sup> )
	<a href="#">loudness_prueba.sce</a>	Prueba de la función loudness

<b>Tipo</b>	<b>Función, script o archivo</b>	<b>Descripción</b>
	<a href="#">critband_data.sce</a>	Datos sobre bandas críticas
	<a href="#">tonotopic.sce</a>	Ánalisis de la curva tonotópica que relaciona las frecuencias de máxima oscilación en la cóclea con la posición.
	<a href="#">ratio_pitch.sce</a>	Parametrización de la curva de altura psicoacústica según Zwicker y Fastl
	<a href="#">Modelo circuitual de la cóclea.pdf</a>	Artículo sobre un modelo eléctrico de la cóclea que permite simular las ondas viajeras
	<a href="#">cochlea_model.sci</a>	Modelo de la cóclea
	<a href="#">cochlea_model_prueba.sce</a>	Prueba del modelo de la cóclea
	<a href="#">cochlea_travelling_wave.sci</a>	Simula mediante un modelo aproximado las ondas viajeras en la cóclea
	<a href="#">animate_cochlea.sce</a>	Muestra una animación de la membrana basilar de la cóclea ante un sonido formado por uno o más armónicos o parciales.
	<a href="#">hrtf.sci</a>	Aplica una función de transferencia relativa a la cabeza
	<a href="#">envelope_corr.sci</a>	Obtiene las correlaciones entre los espectros de las envolventes espectrales procedentes del filtrado por bandas críticas de una señal. Es un indicador útil para estimar si una señal tiene o no rasgos físicos compatibles con la posesión de contenido semántico
	<a href="#">envelope_corr_test.sci</a>	Prueba de la función anterior
MIDI	<a href="#">midi2eventmat.sci</a>	Extrae los eventos MIDI de un archivo MIDI y los entrega tabulados
	<a href="#">eventmat2midi.sci</a>	Convierte una tabla de eventos MIDI en un archivo MIDI. Es útil para generar música algorítmicamente y luego reproducirla en cualquier secuenciador
	<a href="#">ticks2time.sci</a>	Convierte ticks MIDI a tiempo
	<a href="#">deltatime.sci</a>	Obtiene los delta times en un archivo MIDI
Probabilidad y estadística	<a href="#">pdf.sci</a>	Calcula las curvas de densidad de probabilidad para varias distribuciones (normal, chi cuadrado, t de Student y F de Snedecor)
	<a href="#">poisson.sci</a>	Simula los instantes de ocurrencia de eventos de un proceso de Poisson aleatorio. Permite simular eventos que sobrevienen aleatoriamente con un promedio de lambda eventos por segundo
	<a href="#">alpha_rice.sci</a>	Obtiene el parámetro alfa de la distribución de Rice aplicable a la distribución de picos del ruido de banda
	<a href="#">rice_distribution.sci</a>	Distribución de Rice
	<a href="#">zerocross.sci</a>	Detector de cruces por cero

<b>Tipo</b>	<b>Función, script o archivo</b>	<b>Descripción</b>
	<a href="#"><u>localmax.sci</u></a>	Obtiene los máximos locales de una señal
	<a href="#"><u>localmin.sci</u></a>	Obtiene los mínimos locales de una señal
	<a href="#"><u>cummean.sci</u></a>	Promedio acumulativo. Sirve para determinar tiempo de estabilización de un promedio
	<a href="#"><u>polyfit.sci</u></a>	Permite aproximar una curva por polinomios. En la última versión de Scilab se incorporó una función que hace lo mismo y que antes no existía
	<a href="#"><u>smooth_indicator.sci</u></a>	Aproximación analítica y suave a la función indicatriz de un intervalo (1 en el intervalo y 0 fuera del intervalo)
	<a href="#"><u>smooth_curve.sci</u></a>	Suaviza una curva empírica mediante splines por mínimos cuadrados. El resultado se parece bastante al suavizado manual visual.
	<a href="#"><u>animate.sci</u></a>	Permite animar una función que depende de un parámetro a medida que el parámetro evoluciona en el tiempo. Por ejemplo si en una senoide se hace variar la fase, se vería una onda que se desplaza a lo largo del eje x
	<a href="#"><u>cvc_list.sce</u></a>	Generador de listas de sílabas CVC (consonante vocal consonante) con una probabilidad dada de cada fonema

2025-  
10-08 2.6K  
03:48

[filter\\_imp\\_resp\\_prueba.sci](#)

2025-  
10-08 6.1K  
03:19

 [filter\\_imp\\_resp.sci](#)