Normalization of Power Spectral Densities.

Andrew J. Barbour

February 5, 2013

Abstract

A vast and deep pool of literature exists on the subject of spectral analysis; wading through it can obscure even the most fundamental concepts to the inexperienced practitioner. Appropriate interpretation of spectral analyses depends crucially on the normalization used, and here we outline the normalization used by `rlpSpec`, namely the **single-sided power spectral density** (PSD). We briefly outline the background mathematics, present an example from scratch, and compare the results with the normalization used by the spectrum estimator included in the base distribution of R: `stats::spectrum`.

Contents

1 Background

There can often be confusion about the different quantities used in spectral analysis[1], partly due to myriad nomenclature within the incredibly vast literature on the subject. Commonly one finds similarly sounding phrases, including "amplitude spectrum", "energy spectral density", "power", "power spectra", and even "spectra". These all mean *something*, but are rarely equivalent, and can be used improperly.

Let us, for the sake of brevity, assume we are in the time domain, and we are considering a discrete stationary signal x of length N, having a Discrete Fourier Transform $\mathfrak{F}\{f\}$ represented by$X$. The **amplitude spectrum** of this transform pair is simply the amplitude of X, or mod$\{X\}$, which we will denote this as ${}^{(\mathrm{A})}s$. This transform pair's corresponding **phase spectrum** is the phase angle of X, or arg$\{X\}$, denoted by ${}^{(\phi)}s$.

How do we interpret the quantities

Table with equivalent expressions

Table 1: Representors of various spectral quantities.

Expression	Representing	Equivalent expressions
x	stationary timeseries	
X	Fourier transform of x	$\mathfrak{F}\{x\}$
${}^{(\mathrm{A})}s$	**amplitude spectrum** of x	$\lvert X\rvert$ or $\mathrm{mod}\{X\}$
${}^{(\phi)}s$	**phase spectrum** of x	$\arg\{X\}$
${}^{(\mathrm{E})}S$	**energy spectral density** of x	$\lvert X\rvert^2 = \mathrm{mod}\{X \star X^*\} = \mathrm{mod}\{X^* \star X\}$
${}^{(\mathrm{P})}S$	**power spectral density** of x	$\mathcal{E}\{\lvert X\rvert^2\} = \lvert X\rvert^2/N$

2 DCA

To fix the first problem we define $X_T(t)$ as the process $X(t)$ on the finite interval $(-T/2, T/2)$:

$$X_T(t) = \begin{cases} X(t) & -T/2 \leq t \leq T/2 \\ 0 & \text{otherwise.} \end{cases}$$

This is not stationary, but that is all right for now. Any particular realization of this process has a bounded 2-norm and thus has an ordinary Fourier transform:

$$\tilde{X}_T(f) = \mathcal{F}[X_T(t)] = \int_{-\infty}^{\infty} X_T(t) e^{-2\pi i f t}\, dt = \int_{-T/2}^{T/2} X(t) e^{-2\pi i f t}\, dt$$

[1] A nice illustration of the type of confusion common in spectral analyses of confusion is found in this thread on `R-help`:
`http://r.789695.n4.nabble.com/Re-How-do-I-normalize-a-PSD-td792902.html`

But $\tilde{X}_T$ is still a random function of f; to remove this, we we find its squared magnitude and take the expected value: $\mathcal{E}[|\tilde{X}_T(f)|^2]$.

We then let T tend to infinity; but we can easily see that if we do this, this expected value would grow to infinity – and so we divide it by the interval length $2T$ to tame the growth. Putting all this together gives us a function of frequency

$$\begin{aligned} S(f) &= \lim_{T\to\infty} \frac{1}{T}\mathcal{E}[|\tilde{X}_T(f)|^2] \\ &= \lim_{T\to\infty} \mathcal{E}\left[\frac{1}{T}\left|\int_{-T/2}^{T/2} X(t)e^{-2\pi i f t}\,dt\right|^2\right] \end{aligned} \tag{1}$$

Equation (1) defines the power spectral density; it can be shown that $S(f)$ exists for all stationary processes X with zero mean and a bounded variance. It is, obviously, real and non-negative.

2.1 norm

You should be aware that equation (1) defines a particular normalization for the PSD: one that is not always used, and indeed is probably the less common one. Equation (1) defines what is called the **two-sided** PSD, because in it we allow f to run from $-\infty$ to ∞. As with the Fourier tranform of a real function, when X is real the power spectrum $S(f)$ is even: we only need the values for $f \geq 0$. What is used in this case is usually the **one-sided** PSD, which is given by $2S(f)$ for $f \geq 0$; we explain the factor of two below. In using other people's spectra, you should be aware that the spectrum, unspecified, might be either the two-sided spectrum shown for positive f only, or the one-sided spectrum – and you should, even at the risk of being accused of pedantry, specify which one you have used.

Looking at (1) we can observe that $S(f)$ at any particular f is obtained from products of X with itself, so it is related only to the second order moment of X – no third order moments are involved. We already introduced a function involving second-order moments of X, namely the autocovariance $R(\tau$, defined in equations (**??**) and (). Does $S(f)$ provide independent information about X, or is there a connection between $R(\tau)$ and $S(f)$? Somewhat surprisingly the answer is that the functions $R(\tau)$ and $S(f)$ contain *exactly* the same information, because $S(f)$ is just the Fourier transform of $R(\tau)$

$$S(f) = \mathcal{F}[R(t)] = \int_{-\infty}^{\infty} R(t)e^{-2\pi i f t}\,dt \tag{2}$$

Equation (2) is often used as an alternative definition of the PSD, not least because it is (as we will see) more useful than equation (1) for proving theorems about the PSD.

For real data, $R(\tau)$ is not just even (as it always is) but also real. In that case we can write

$$S(f) = 1/2 \int_0^\infty R(t) \cos(-2\pi f t)\, dt \tag{3}$$

where $S(f)$ is the two-sided spectrum; the $1/2$ absorbs a factor of two produced when we change the limits of the integral. If we used the one-sided spectrum (as makes sense for a real series) we would omit the $1/2$; for generality we will use the definition given by equation (2).

Before we establish the truth of (2) we note a few consequences of it. Since $R(t)$ is a real even function of t, (2) implies that $S(f)$ is also a real and even function in f. But the fact that $S(f)$ must be non-negative puts severe restrictions on what functions $R(t)$ might be autocovariances; clearly not every even function with a Fourier transform will have a positive Fourier transform.

The inverse transform of (2) is

$$R(t) = \int_{-\infty}^\infty S(f) e^{2\pi i f t}\, df \tag{4}$$

and recall from the definition of $R(t)$ that

$$R(0) = \mathcal{E}[X(t)X(t)] = \mathcal{V}[X] = \sigma^2$$

remembering that X is a zero-mean process. Setting $t = 0$ in (2) gives an important result:

$$\sigma^2 = \int_{-\infty}^\infty S(f)\, df \tag{5}$$

That is, *the area under the power spectrum is the variance of the process.* It is to preserve this property that we double $S(f)$ if we use the one-sided PSD; then the integrated area is again the variance even though we we integrate $S(f)$ only over positive frequencies.

Now we verify (2); the argument follows the lines of the one given in Section **??**, except that here we are considering random functions, and there we were considering conventional functions. We start with the squared magnitude of the Fourier transform of the windowed function X_T:

$$|\tilde{X}_T|^2 = \tilde{X}_T \tilde{X}_T^* \tag{6}$$

Recall that the Fourier transform of a convolution is the product of the Fourier transforms; and, since X_T is real

$$\tilde{X}_T^*(f) = \int_{-\infty}^\infty X_T(t) e^{2\pi i f t}\, dt = \int_{-\infty}^\infty X_T(-t) e^{-2\pi i f t}\, dt = \mathcal{F}[X_T(-t)] \tag{7}$$

Combining the convolution theorem with (7) and (6), we have

$$|\tilde{X}_T|^2 = \mathcal{F}[X_T(t) * X_T(-t)] = \mathcal{F}\left[\int_{-\infty}^{\infty} X_T(s) X_T(s-t)\, ds\right] \tag{8}$$

In (1) we have normalized by the interval T, so we put that into the definition of another function:

$$R_T(t) = \frac{1}{T} X_T(t) * X_T(-t) = \frac{1}{T}\int_{-\infty}^{\infty} X_T(s) X_T(s-t)\, ds \tag{9}$$

Then by (8), the Fourier transform of R_T is

$$\mathcal{F}[R_T] = \int_{-\infty}^{\infty} e^{-2\pi i f t} R_T(t)\, dt = \frac{1}{T}\mathcal{F}[X_T(t) * X_T(-t)] = \frac{|\tilde{X}_T|^2}{T} \tag{10}$$

Our definition of the PSD is (1); let us plug (10) into that so that we get

$$S(f) = \lim_{T\to\infty} \mathcal{E}\left[\frac{|\tilde{X}_T|^2}{T}\right] = \lim_{T\to\infty}\int_{-\infty}^{\infty} e^{-2\pi i f t}\mathcal{E}[R_T(t)]\, dt \tag{11}$$

From (9) we see that $R_T(t)$ is even in t, so we can always write $R_T(t) = R_T(|t|)$; in the following we will assume $t \geq 0$ and then replace t by $|t|$ at the end. We know $X_T(s)$ vanishes outside the interval $(-T/2, T/2)$ and therefore the integrand of (9) must vanish when $s > T/2$ or when $|s - t| > T/2$; see Figure **??**. Therefore we can reduce the interval of integration in (9) to be over $(-T/2 + t,\, T/2)$ instead of the whole real line. Also observe that, once $t > T/2$, the nonzero sections cease to overlap, and the integrand is identically zero.

These considerations lead to the result

$$R_T(t) = \begin{cases} \frac{1}{T}\int_{-T/2+t}^{T/2} X_T(s) X_T(s-t)\, ds & 0 \leq t < T/2 \\ 0 & t \geq T/2 \end{cases} \tag{12}$$

Further simplifications ensue when we take the expected value, as dictated by (11); for the segment $0 \leq t < T/2$

$$\mathcal{E}[R_T(t)] = \frac{1}{T}\int_{-T/2+t}^{T/2} \mathcal{E}[X_T(s) X_T(s-t)]\, ds = \frac{1}{T}\int_{-T/2+t}^{T/2} R(-t)\, ds$$

where we have, finally, introduced the actual autocovariance of the process, $R(t)$. Since $R(-t) = R(t)$, which is independent of s, we can evaluate the s integral explicitly:

$$\mathcal{E}[R_T(t)] = \frac{R(t)}{T}\int_{-T/2+t}^{T/2} 1 \cdot ds = R(t)\left[1 - \frac{t}{T}\right] \qquad \text{for} \quad 0 \leq t \leq T \tag{13}$$

From (12) $\mathcal{E}[R_T(t)] = 0$ when $t \geq T$. Recalling that R_T is even, we can write the negative t behavior from $R_T(t) = R_T(-t)$, and obtain the following complete description for the expected value of R_T:

$$\mathcal{E}[R_T(t)] = R(t)\Lambda_T(t) \tag{14}$$

where

$$\Lambda_T(t) = \left\{ \begin{array}{cc} 1 - |t|/T & |t| \leq T \\ 0 & |t| > T \end{array} \right\}$$

which is just a triangle function of unit height and width $2T$.

Substituting (14) into (11) gives us the following very plausible expression for the PSD:

$$S(f) = \lim_{T\to\infty} \mathcal{F}[R(t)\Lambda_T(t)] = \lim_{T\to\infty} \int_{-\infty}^{\infty} e^{-2\pi ift} R(t)\Lambda_T(t)\, dt \tag{15}$$

If we can put the limit in (15) inside the integral, and since $\Lambda_T(s) \to 1$ as $T \to \infty$, we then have the result we claimed: equation (2). This last step is where some care is needed. **?** (pp. ??) uses the Lebesgue Dominated Convergence Theorem, and the further condition that

$$\int_{-\infty}^{\infty} |R(t)|\, dt < \infty \tag{16}$$

to prove that it is permitted to reverse the order of the limit and the integral. Section **??** contains a proof (by Parker) that makes a different set of assumptions about $R(t)$.

3 A from-scratch example: White noise.

A straightforward way to understand normalization in spectral analysis is to analyze a real, stationary series which is normally distributed with known variance, $x = \mathcal{N}(\mu, \sigma^2)$. A fundamental result found in many texts on spectral analysis is

$$\mathrm{var}\{x\} \equiv \sigma_x^2 = \int_{-1/2}^{1/2} {}^{(\mathrm{P})}S(f)df = 2\int_{0}^{1/2} {}^{(\mathrm{P})}S(f)df \tag{17}$$

which says if we integrate the power spectral density over all frequencies we can obtain the variance of the source process. If we have a $\mathcal{N}(0,1)$ process, and assume the sampling interval is once per second, we should expect a flat spectrum of 2 units^2/Nyquist across all frequencies $[0, 0.5]$ so that the area under the spectrum is equal to one.

We can illustrate this with a few lines of code. First, generate a series, and then find its iscrete Fourier Transform (DFT)[2].

[2] A proper DFT is normalized by the length of the series; however, most DFT calculators (including `stats::fft`) eschew this normalization for efficiency's sake.

```
set.seed(1234)
N <- 256
x <- rnorm(N, mean = 0, sd = 1)
xv <- var(x)
X <- fft(x)
class(X)

## [1] "complex"

length(X)

## [1] 256
```

We can easily find the amplitude and phase response:

```
Sa <- Mod(X)  # Amplitude spectrum
Sp <- Arg(X)  # Phase spectrum
```

followed by equivalent energy spectral density calculations[3]

```
XC <- Conj(X)
all.equal(Se <- Sa^2, Se_2 <- Mod(XC * X), Se_2R <- Mod(X * XC))

## [1] TRUE
```

The single-sided power spectral density (PSD) estimates follow once the Nyquist frequency is set; this is defined as half the sampling rate[4].

```
fsamp <- 1  # sampling freq, Hz
fNyq <- fsamp/2  # nyquist
Nf <- N/2
nyfreqs <- seq.int(from = 0, to = fNyq, length.out = Nf)
S <- Se[1:Nf] * 2/N  # Finally, the PSD!
```

An estimate of the integrated spectrum should roughly equal the known variance. Figure 3 plots the PSD of our white noise series; it also shows the mean value of the PSD[5], from which we can perform a variance–normalization test:

[3] Note the equivalence between the complex conjugate based estimates.

[4] Although a white noise process is not strictly bandlimited, we will use it to demonstrate differences in normalization.

[5] Estimates for the PSD of a white noise series are approximately log-normally distributed; hence, a simple mean value is highly biased estimator.

```
plot(nyfreqs, S, type = "h", xlab = "Nyquist frequency", ylab = "units**2 / freq")
print(c(mSn <- mean(S), mSm <- median(S)))

## [1] 2.034 1.242

abline(h = c(mSn, mSm), lwd = 2, lty = c(2, 3), col = "red")
```

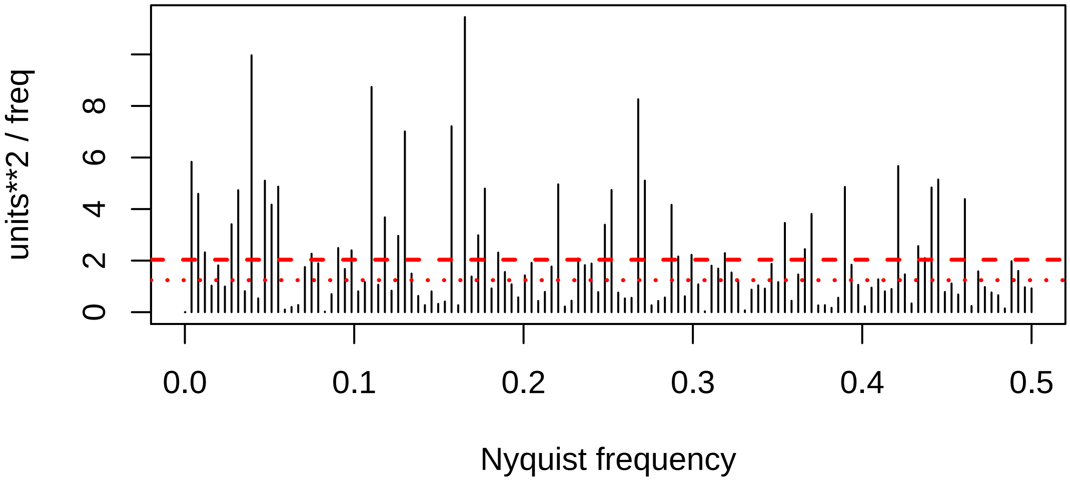


Figure 1: Power spectral density estimates for a single realization of a $\mathcal{N}(0,1)$ process in linear units. The dashed line shows the mean spectral level and the dotted line shows the median spectral level; these can be used to find the integrated spectrum and test normalization.

```
test_norm <- function(sval, nyq, xvar) {
    svar <- sval * nyq
    return(svar/xvar)
}
print(xv_1 <- test_norm(mSn, fNyq, xv))

## [1] 0.9933

xv_2 <- sum(S)/Nf * fNyq/xv  # an alternate test
```

```
all.equal(xv_1, xv_2)

## [1] TRUE
```

But what if the sampling frequency `fsamp` changes? An obvious change will be the actual Nyquist frequency, which means the variance–normalization test will fail if the PSD estimates are not re-scaled. We simply re-scale the frequencies and PSD with the sampling rate to obtain the properly-normalized spectra.

```
fsamp <- 20
fNyq <- fsamp/2
freqs <- fsamp * nyfreqs
Snew <- S/fsamp
```

To compare the scalings it is helpful to instead show the spectral values in decibels (relative to 1 units^2/frequency).

4 Normalization used in `stats::spectrum`

We wish to compare the normalizations used by other PSD estimation programs; these are summarized in 3.

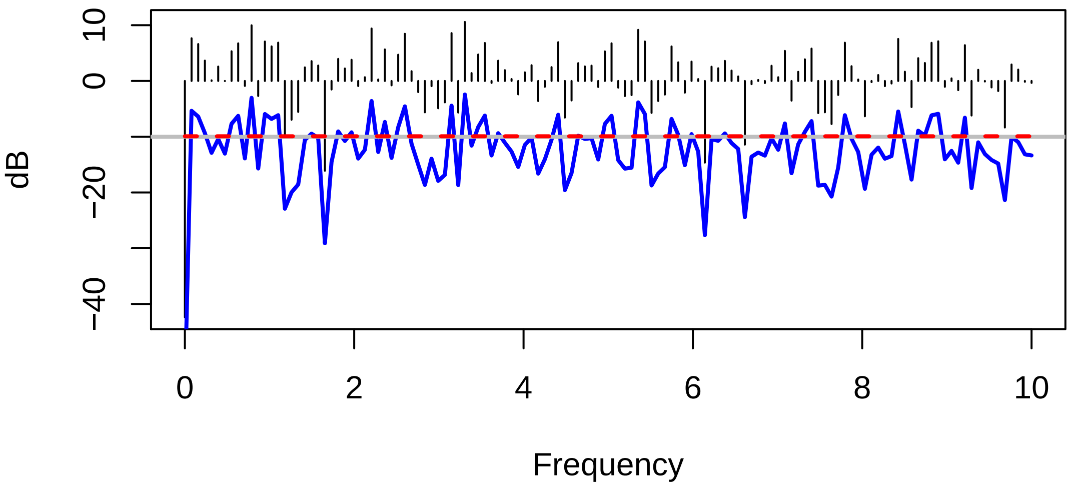
Table 2: A comparison of power spectral density estimators in R, excluding extensions which only estimate raw-periodograms. Normalizations are shown as either "single" or "double" for either single- or double-sided spectra, and "various" if there are multiple, optional normalizations. A (∗) denotes the default for a function having an option for either single or double.

Function	Namespace	Sine m.t.?	Adaptive?	Norm.	Reference
`mtapspec`	`RSEIS`	Yes	No	various	Lees and Park (1995)
`pspectrum`	`rlpSpec`	Yes	Yes	single	Parker and Barbour (2013)
`spectrum`	`stats`	No	No	double	R Core Team (2012)
`spec.mtm`	`multitaper`	Yes	Yes	double	Rahim and Burr (2012)
`SDF`	`sapa`	Yes	No	single*	Percival and Walden (1993)

4.1

Included in the core distribution of R is `stats::spectrum`, which accesses `stats::spec.ar` or `stats::spec.pgram` for either parametric and non-parametric estimation, respectively.

```
# decibel function
dB <- function(y) 10 * log10(y)
# and some plots...
plot(freqs, dB(S), type = "h", xlab = "Frequency", ylab = "dB")
lines(freqs, dB(Snew), col = "blue", lwd = 2)
abline(h = dB(1/fNyq), col = "grey", lwd = 2)
mSn <- mean(Snew)
lines(c(0, fNyq), rep(dB(mSn), 2), lwd = 2, lty = 2, col = "red")
```



```
# finally, test variance.
test_norm(mSn, fNyq, xv)

## [1] 0.9933
```

Figure 2: Rescaled PSD estimates for a single realization of a $\mathcal{N}(0, 1)$ process with a sampling rate of 20 s^{-1} rather than 1 s^{-1} as from before. The dashed line shows the mean (rescaled) spectral level, and the grey line shows the predicted mean value from the Nyquist frequency.

The user can optionally apply a single cosine taper, and/or a smoothing kernel. For this discussion we compare to `spec.pgram`.

`spec.pgram` assumes the sampling frequency for the input series is 1, and normalizes accordingly; however, sampling information used be included by creating a `ts` object from the series prior to spectrum estimation:

```
fsamp <- 20
xt <- ts(x, frequency = fsamp)
pgram20 <- spec.pgram(xt, pad = 1, taper = 0, plot = FALSE)
pgram01 <- spec.pgram(ts(xt, frequency = 1), pad = 1, taper = 0, plot = FALSE)
```

A first order question is obviously whether these spectra pass our variance-normalization test: they do not, but only by a factor of two (too small):

```
test_norm(mean(pgram01$spec), 0.5, xv)

## [1] 0.4845

test_norm(mean(pgram20$spec), 10, xv)

## [1] 0.4845
```

But why? The program assumes normalization for a **double-sided spectrum**, which conflicts with our definition of the one-sided spectrum by a factor of two. We can illustrate this with the following example:

```
psd1 <- spec.pgram(x, plot = FALSE)
psd2 <- spec.pgram(xc <- complex(real = x, imag = x), plot = FALSE, demean = TRUE)
mx <- mean(Mod(x))
mxc <- mean(Mod(xc))
(mxc/mx)^2

## [1] 2

mean((psd2$spec/psd1$spec))

## [1] 2
```

This means that unless we are interested in analyzing complex timeseries, we need only multiply by two for properly normalized spectra using `spectrum`, assuming the sampling information is included in the series.

```
plot(pgram20, log = "dB", ylim = 36 * c(-1, 0.3))
plot(pgram01, log = "dB", add = TRUE, col = "red")
abline(h = dB(c(1, 1/2/1, 1/2/20)), col = c("grey", "red", "black"))
```

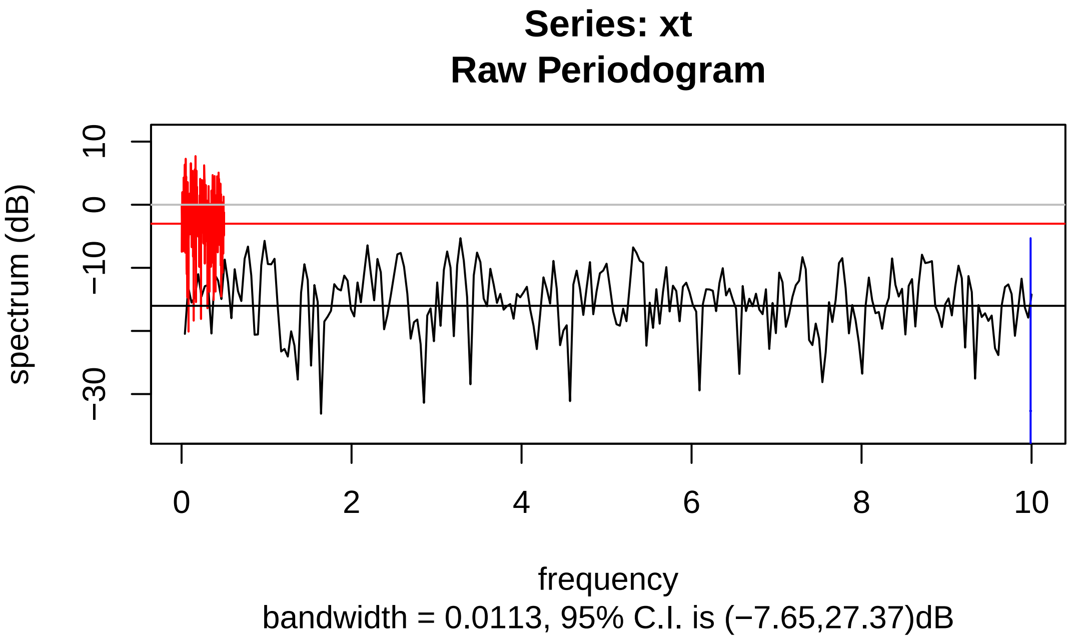


Figure 3: `spec.pgram`

5 Other PSD estimators

The suite of extensions to base R which have similar functionality is relatively limited; however, there are at least three with can produce sophisticated PSD estimates. We have summarized the available functions in Table 3 so far as we know[6].

References

Lees, J. M. and Park, J. (1995). Multiple-taper spectral analysis: A stand-alone C-subroutine. *Computers & Geosciences*, 21(2):199–236.

[6] As of this writing (Feb 2013), `sapa` appears to be orphaned.

Table 3: A comparison of power spectral density estimators in R, excluding extensions which only estimate raw-periodograms. Normalizations are shown as either "single" or "double" for either single- or double-sided spectra, and "various" if there are multiple, optional normalizations. A (∗) denotes the default for a function having an option for either single or double.

FUNCTION	NAMESPACE	SINE M.T.?	ADAPTIVE?	NORM.	REFERENCE
`mtapspec`	`RSEIS`	YES	NO	various	Lees and Park (1995)
`pspectrum`	`rlpSpec`	YES	YES	single	Parker and Barbour (2013)
`spectrum`	`stats`	NO	NO	double	R Core Team (2012)
`spec.mtm`	`multitaper`	YES	YES	double	Rahim and Burr (2012)
`SDF`	`sapa`	YES	NO	single*	Percival and Walden (1993)

Parker, R. L. and Barbour, A. J. (2013). *rlpSpec: Adaptive, sine-multitaper power spectral density estimation.* R package version 0.2-0.

Percival, D. and Walden, A. (1993). *Spectral analysis for physical applications.* Cambridge University Press.

R Core Team (2012). *R: A Language and Environment for Statistical Computing.* R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-07-0.

Rahim, K. and Burr, W. (2012). *multitaper: Multitaper Spectral Analysis.* R package version 1.0-2.

Index