Chapter 8. Smoothing in the time and frequency domains

Objectives

1. Estimating a nonparametric trend from a time series is known as smoothing. We will review some standard smoothing methods.
2. We can also smooth the periodogram to estimate a spectral density.
3. Many smoothers can be represented as linear filters. We will see that the statistical properties of linear filters for dependent (time-domain) stationary models can be conveniently studied in the frequency domain.

A motivating example

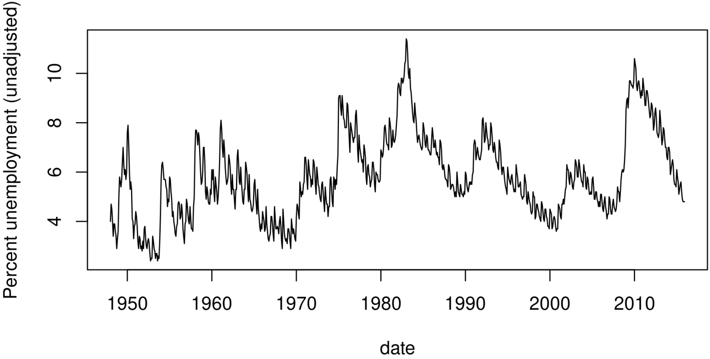
- The economy fluctuates between periods of rapid expansion and periods of slower growth or contraction.
- High unemployment is one of the most visible signs of a dysfunctional economy, in which labor is under-utilized, leading to hardships for many individuals and communities.
- Economists, politicians, businesspeople and the general public therefore have an interest in understanding fluctuations in unemployment.
- Economists try to distinguish between fundamental structural changes in the economy and the shorter-term cyclical booms and busts that appear to be a natural part of capitalist business activity.
- Monthly US unemployment figures are published by the Bureau of Labor Statistics (BLS) (`data.bls.gov/timeseries/LNU04000000`).
- Measuring unemployment has subtleties, but these are not our immediate focus.

```
system("head unadjusted_unemployment.csv",intern=TRUE)

##  [1] "# Data extracted on: February 4, 2016 (10:06:56 AM)"
##  [2] "# from http://data.bls.gov/timeseries/LNU04000000"
##  [3] "# Labor Force Statistics from the Current Population Survey
##  [4] "# Not Seasonally Adjusted"
##  [5] "# Series title:        (Unadj) Unemployment Rate"
##  [6] "# Labor force status:  Unemployment rate"
##  [7] "# Type of data:        Percent or rate"
##  [8] "# Age:                 16 years and over"
##  [9] "Year,Jan,Feb,Mar,Apr,May,Jun,Jul,Aug,Sep,Oct,Nov,Dec"
## [10] "1948,4.0,4.7,4.5,4.0,3.4,3.9,3.9,3.6,3.4,2.9,3.3,3.6"

U1 <- read.table(file="unadjusted_unemployment.csv",
  sep=",",header=TRUE)
head(U1,3)

##   Year Jan Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec
## 1 1948 4.0 4.7 4.5 4.0 3.4 3.9 3.9 3.6 3.4 2.9 3.3 3.6
## 2 1949 5.0 5.8 5.6 5.4 5.7 6.4 7.0 6.3 5.9 6.1 5.7 6.0
## 3 1950 7.6 7.9 7.1 6.0 5.3 5.6 5.3 4.1 4.0 3.3 3.8 3.9
```

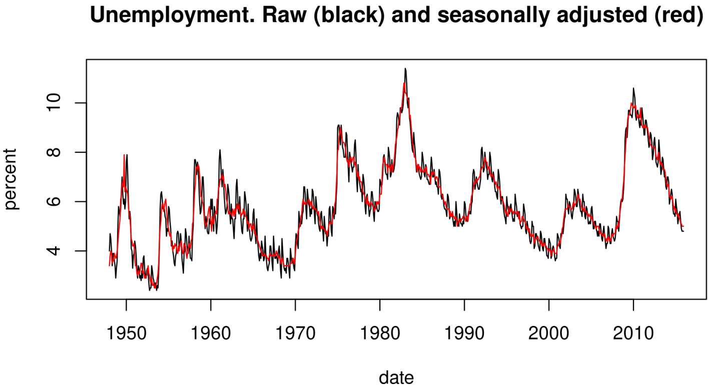
The data are in a table, and we want a time series.

```
u1 <- t(as.matrix(U1[2:13]))
dim(u1) <- NULL
date <- seq(from=1948,length=length(u1),by=1/12)
plot(date,u1,type="l",ylab="Percent unemployment (unadjusted)")
```

Question 8.1. Explain how this code works.



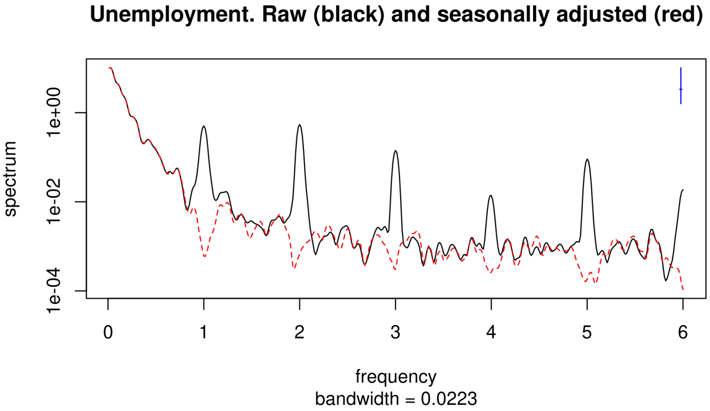
- We see seasonal variation, and perhaps we see business cycles on top of a slower trend.
- The seasonal variation looks like an additive effect, say an annual fluctation with amplitude around 1 percentage point.
- Sometimes, we may prefer to look at monthly seasonally adjusted unemployment (`data.bls.gov/timeseries/LNS14000000`).



We may be curious how the Bureau of Labor Statistics adjusts the data, and if this introduces any artifacts that a careful statistician should be

- To help understand the seasonal adjustment, we look at what it does to the smoothed periodogram.
- Using the ts class we can tell R the units of time.

```
u1_ts <- ts(u1,start=1948,frequency=12)
u2_ts <- ts(u2,start=1948,frequency=12)
spectrum(ts.union(u1_ts,u2_ts),spans=c(3,5,3),
  main="Unemployment. Raw (black) and seasonally adjusted (red)")
```

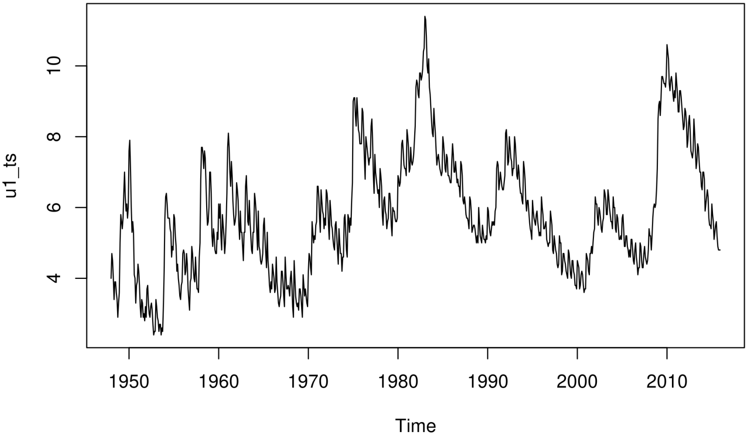


Question 8.2. What are the x-axis units?

Question 8.3. Comment on what you learn from comparing these smoothed periodograms.

The ts class can also be useful for helping R choose other plotting options in a way appriate for time series. For example,

```
plot(u1_ts)
```



- Note: For a report, we should add units to plots. Extra details (like bandwith in the periodogram plot) should be explained or removed.

The transfer (or frequency response) function

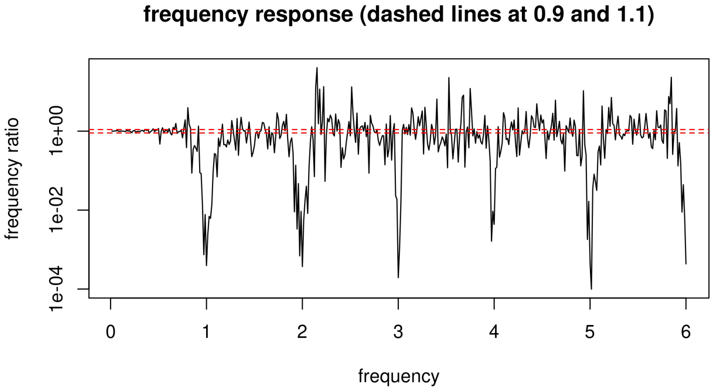
- The ratio of the periodograms of the smoothed and unsmoothed time series is called the **transfer function** or **frequency response function** of the smoother.
- We can infer the frequency response of the smoother used by Bureau of Labor Statistics to deseasonalize the unemployment data.

```r
s <- spectrum(ts.union(u1_ts,u2_ts),plot=FALSE)
```

- We need to figure out how to extract the bits we need from `s`

```r
names(s)

##  [1] "freq"      "spec"      "coh"       "phase"
##  [5] "kernel"    "df"        "bandwidth" "n.used"
##  [9] "orig.n"    "series"    "snames"    "method"
## [13] "taper"     "pad"       "detrend"   "demean"
```

```
plot(s$freq,s$spec[,2]/s$spec[,1],type="l",log="y",
  ylab="frequency ratio", xlab="frequency",
  main="frequency response (dashed lines at 0.9 and 1.1)")
abline(h=c(0.9,1.1),lty="dashed",col="red")
```

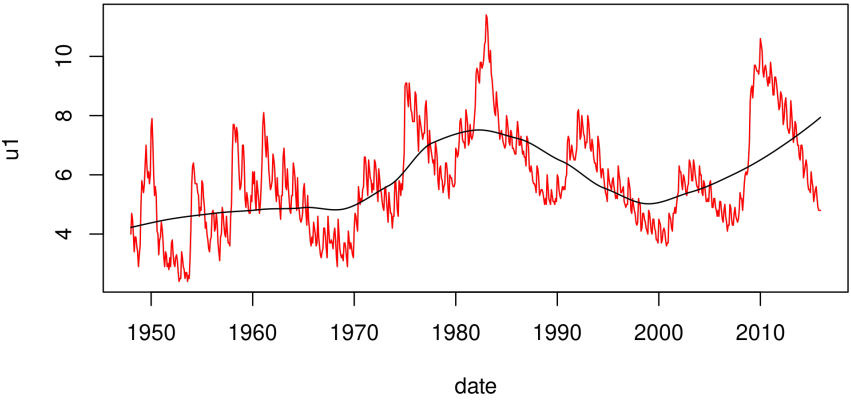


Question 8.4. What do you learn from this frequency response plot?

Estimating trend by Loess smoothing

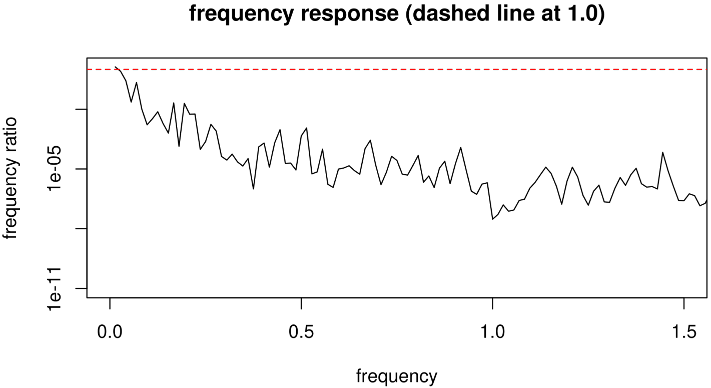
- Loess is a **Local linear regression** approach (perhaps an acronym for LOcal Estimation by Smoothing?)
- At each point in time, Loess makes a linear regression (e.g., fit a constant, linear or quadratic polynomial) using only points close in time.
- We can imagine a moving window of points included in the regression.
- `loess` is an R implementation, with the fraction of points included in the moving window being scaled by the `span` argument.
- Let's choose a value of the span that visually separates long term trend from business cycle.

```
u1_loess <- loess(u1~date,span=0.5)
u1_loess <- loess(u1~date,span=0.5)
plot(date,u1,type="l",col="red")
lines(u1_loess$x,u1_loess$fitted,type="l")
```



Now, we compute the frequency response function for what we have done.

```
s2 <- spectrum(ts.union(
  u1_ts,ts(u1_loess$fitted,start=1948,frequency=12)),
  plot=FALSE)
plot(s2$freq,s2$spec[,2]/s$spec[,1],type="l",log="y",
  ylab="frequency ratio", xlab="frequency", xlim=c(0,1.5),
  main="frequency response (dashed line at 1.0)")
abline(h=1,lty="dashed",col="red")
```

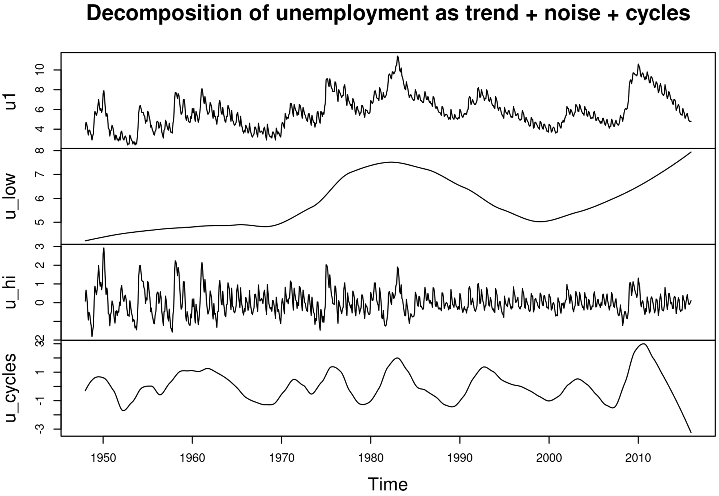


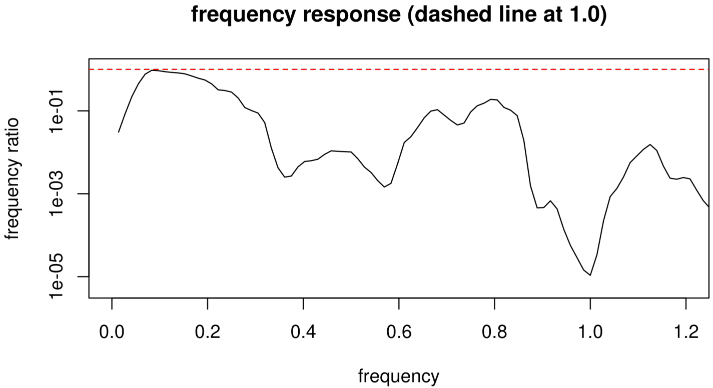
Question 8.5. Describe the frequency domain behavior of this filter.

Extracting business cycles: A band pass filter

- For the unemployment data, high frequency variation might be considered "noise" and low frequency variation might be considered trend.
- A band of mid-range frequencies might be considered to correspond to the business cycle.
- Let's build a smoothing operation in the time domain to extract business cycles, and then look at its frequency response function.

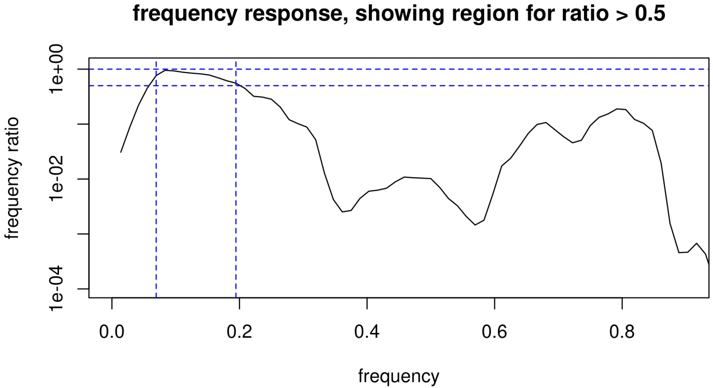
```r
u_low <- ts(loess(u1~date,span=0.5)$fitted,
  start=1948,frequency=12)
u_hi <- ts(u1 - loess(u1~date,span=0.1)$fitted,
  start=1948,frequency=12)
u_cycles <- u1 - u_hi - u_low
plot(ts.union(u1, u_low,u_hi,u_cycles),
  main="Decomposition of unemployment as trend + noise + cycles")
```


Decomposition of unemployment as trend + noise + cycles
u1
4
6
8
10
u_low
5
6
7
8
u_hi
0
1
2
3
u_cycles
-3
-1
1
32
1950
1960
1970
1980
1990
2000
2010
Time



Question 8.6. Describe the frequencies (and corresponding periods) that this decomposition identifies as business cycles. Note: Usually, we should specify units for frequency and period. Here, the units are omitted to give you an exercise!

To help answer this question, let's add some lines to the previous plot



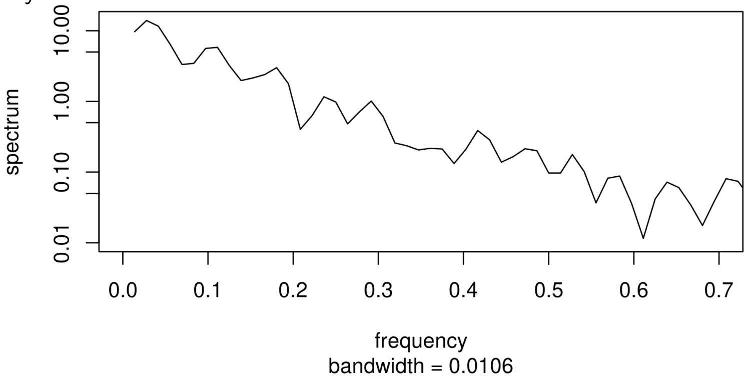
```r
kable(matrix(freq_cycles,nrow=1,
  dimnames=list("frequency",c("low","hi"))),digits=3)
```

	low	hi
frequency	0.069	0.194

Question 8.7. So far as we have opinions on business cycles, use them to comment on this decomposition.

Looking for business cycles

We plot the lower frequencies of a smoothed periodogram for the raw unemployment data to zoom in on frequencies around the business cycle frequency.



Question 8.8. Comment on the evidence for and against the concept of a business cycle in the above figure.

Common smoothers in R

- Above, we have used the **local regression smoother** `loess` but there are other options.
- Our immediate goal is to get practical experience using a smoother and then statistically assessing what we have done.
- You can learn about alternative smoothers, and try them out, if you like.
- `ksmooth` is a **kernel smoother**. The default periodogram smoother in `spectrum` is also a kernel smoother. See `https://en.wikipedia.org/wiki/Kernel_smoother`
- `smooth.spline` is a **spline smoother**. `https://en.wikipedia.org/wiki/Smoothing_spline`

Bandwidth for a smoother

- All these smoothers have some concept of a **bandwidth**, which is a measure of the size of the neighborhood of time points in which data affect the smoothed value at a particular time point.
- The concept of bandwidth is most obvious for kernel smoothers, but exists for other smoothers.
- We usually only interpret bandwidth up to a constant. For a particular smoothing algorithm and software implementation, you learn by experience to interpret the comparative value. Smaller bandwidth means less smoothing.
- Typically, when writing reports, it makes sense to focus on the tuning parameter for the smoother in question, which is not the bandwidth except for a kernel smoother.

Acknowledgments and License

- These notes build on previous versions at `ionides.github.io/531w16` and `ionides.github.io/531w18`.