Filtering and Smoothing

Jared Fisher

Lecture 3b

Announcements

Announcements

- Project Checkpoint 1 was due yesterday. Let me know of any issues.
- Fifth lab section is now open.
- Homework 2 is due next week, Wednesday Feb 17
- Midterm 1 is on Thursday Feb 25

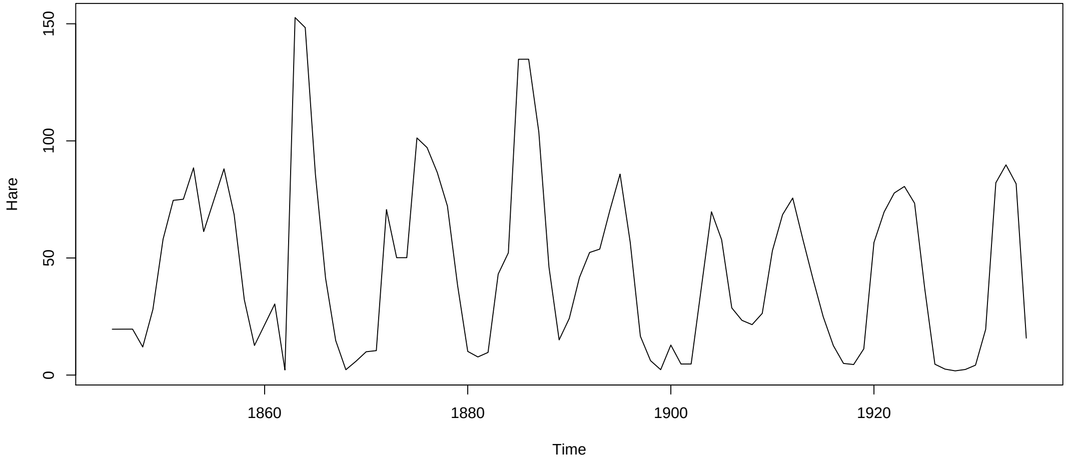
Recap

Full Model

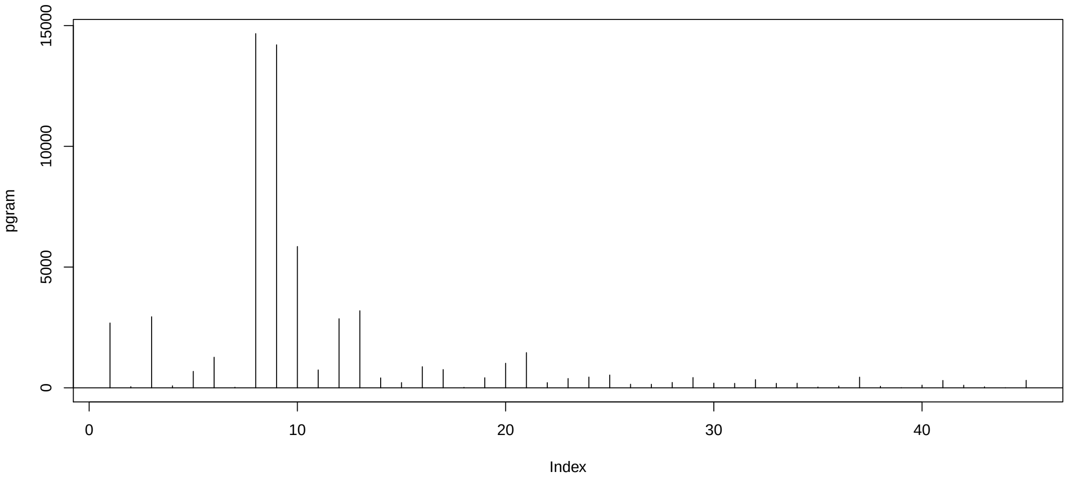
$$Y_t = m_t + s_t + X_t$$

- m_t is the trend (up to now has been deterministic)
- s_t is the seasonal effect (up to now has been deterministic)
- X_t is as stationary process, perhaps white noise
- **Idea:** Remove trend and seasonality, so that residuals $Y_t - \hat{m}_t - \hat{s}_t$ exhibit steady behavior over time, i.e. looks stationary.

Modeling with Sinusoids Example: Hare



Finding Frequencies with Periodogram (n=91)



Definition: Periodogram

For real values data $x_0, \ldots, x_{n-1}$ with DFT $b_0, \ldots, b_{n-1}$ the **periodogram** is defined as

$$I(j/n) = \frac{|b_j|^2}{n} \quad \text{for } j = 1, \ldots, \lfloor n/2 \rfloor$$

Definition: Discrete Fourier Transform

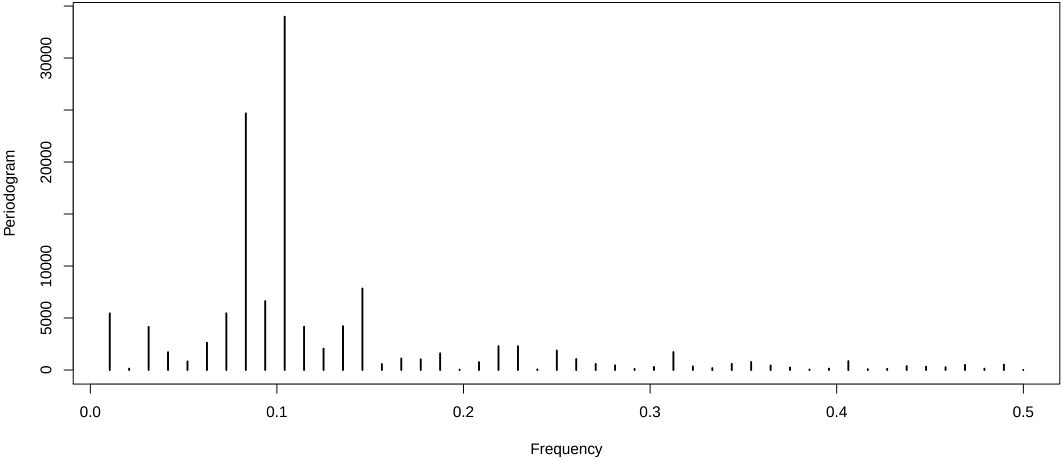
For data $x_0, \ldots, x_{n-1} \in C$ the discrete Fourier transform (DFT) is given by $b_0, \ldots, b_{n-1} \in C$, where

$$b_j = \sum_{t=0}^{n-1} x_t \exp\left(-\frac{2\pi ijt}{n}\right) \text{ for } j = 0, \ldots, n-1.$$

(In R, the DFT is calculated by the function *fft()*.)

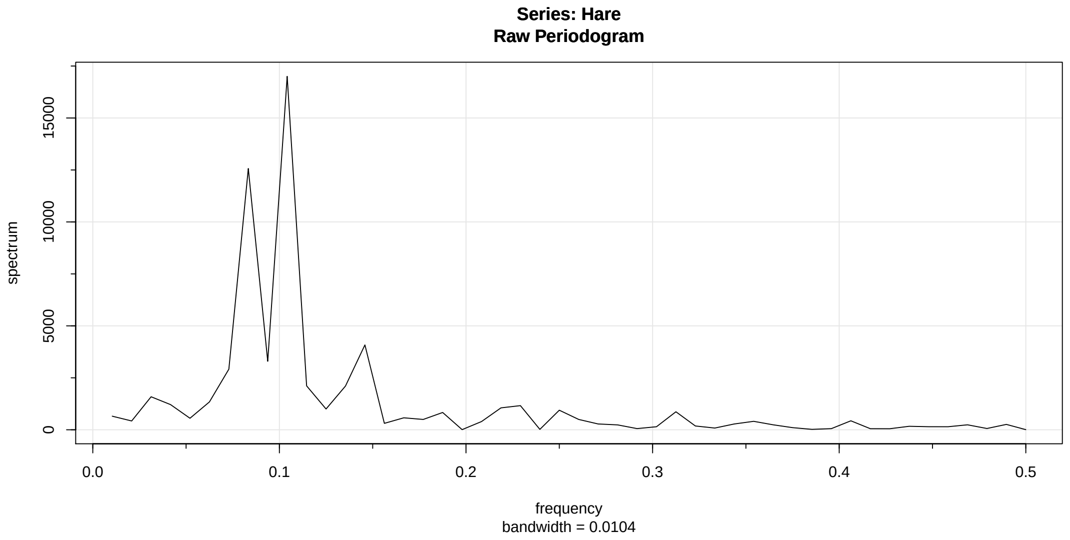
Other Periodogram functions have different # of frequencies

```
TSA::periodogram(Hare)
```



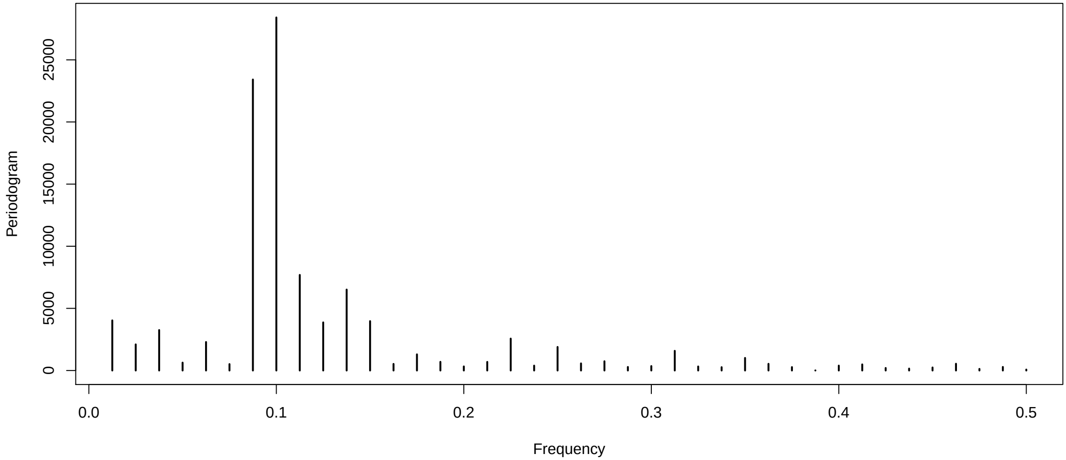
Smoothed Periodogram

```
astsa::mvspec(Hare)
```



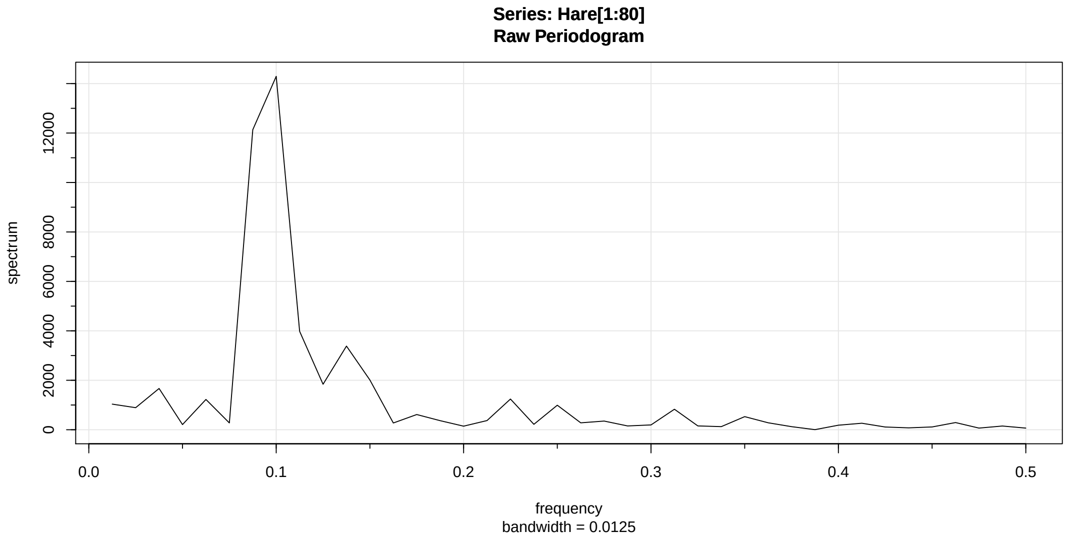
Other Periodogram functions have different # of frequencies

```r
# 80 vs. 81 shouldn't make a big difference, right?
TSA::periodogram(Hare[1:80])
```



Smoothed Periodogram, n-1 points

```r
astsa::mvspec(Hare[1:80])
```



Note on Periodogram functions

- This Hare example shows that there may be one or two dominant frequencies depending on how we partition the frequency domain
- Is there a single frequency of importance and we're just seeing leakage? Or are there two?
- There's not a definitive answer in these plots – that choice is up to you as the modeler/analyst.
- (Perhaps fit both and see which works better! Or it may simply be that one makes more sense than the other)

Filters

Textbook Alignment

Section 4.7 (but ignore the parts on “spectrum”, we’ll cover that later)

Filters

- Now for something different!
- We have modeled m_t and s_t as a **deterministic** functions of time...
- We will relax that now, as our main goal is not to get functions, but to *"pursue stationarity"* in our residuals!
- The general technique of linear time invariant filters: transforming one time series into another.

Definition: Linear Time Invariant Filter

A linear time-invariant filter with coefficients $\{a_j\}$ for $j = \ldots, -2, -1, 0, 1, 2, 3, \ldots$ transforms an input time series $\{U_t\}$ into an output time series $\{V_t\}$ via

$$V_t = \sum_{j=-\infty}^{\infty} a_j U_{t-j}.$$

In the above definition, the coefficients $\{a_j\}$ are often assumed to satisfy $\sum_{j=-\infty}^{\infty} |a_j| < \infty$.

In-class Practice

Let U_t be stationary with

- $E(U_t) = \mu$
- $var(U_t) = \sigma^2$
- $cov(U_t, U_{t+h}) = \gamma(h)$.

For $V_t = \sum_{j=-\infty}^{\infty} a_j U_{t-j}$, evaluate the following in terms of $\mu, \sigma^2, \gamma(h)$.

- $E(V_t)$
- $cov(V_t, V_{t+h})$
- $var(V_t)$
- Take 5 minutes, then we'll do these on the board

Autocovariance of Linear Time Invariant Filter

- Suppose that the input time series $\{U_t\}$ is stationary with autocovariance function γ_U.
- Then for the autocovariance function (ACVF) of $\{V_t\}$ we observe

$$
\begin{aligned}
\gamma_V(h) =& \operatorname{cov}(V_t, V_{t+h}) \\
=& \operatorname{cov}\left(\sum_j a_j U_{t-j}, \sum_k a_k U_{t+h-k}\right) \\
=& \sum_{j,k} a_j a_k \operatorname{cov}(U_{t-j}, U_{t+h-k}) \\
=& \sum_{j,k} a_j a_k \gamma_U(h-k+j).
\end{aligned}
$$

- Note that the above calculation shows also that $\{V_t\}$ is stationary.

Examples

- Particular types of time invariant linear filters we will look at:
- q-step smoothing $\Rightarrow a_j = \frac{1}{2q+1}$ for $|j| \leq q$, $a_j = 0$ otherwise.
- Exponential smoothing $\Rightarrow a_j \propto \alpha^j$ for $j > 0$, $\alpha \in (0, 1)$, and $a_j = 0$ otherwise.
- First differencing $\Rightarrow a_0 = 1$ and $a_1 = -1$, $a_j = 0$ otherwise.
- These filters act very differently; the first two estimate trend while the other eliminates it.

Smoothing

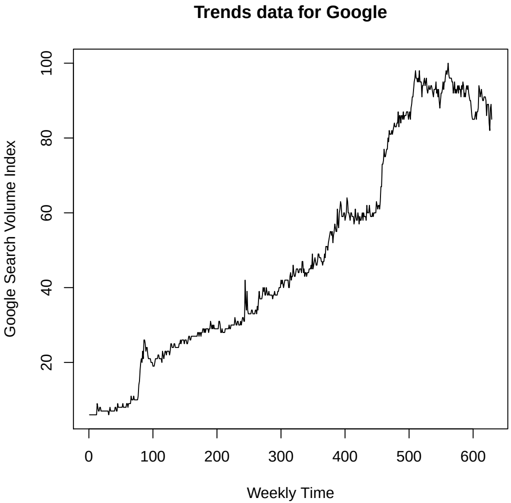
Smoothing

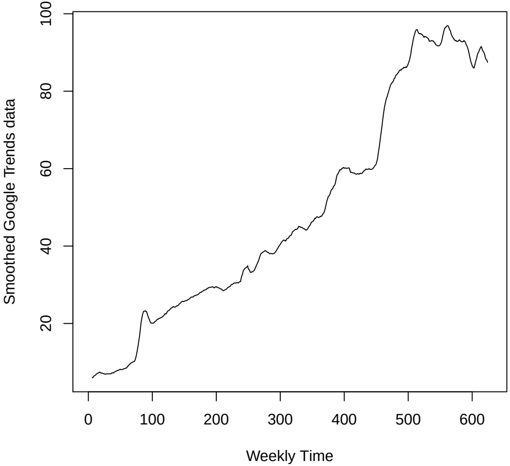
Assume simple trend model $Y_t = m_t + W_t$, we estimate m_t by averaging in a neighborhood $[t-q, t+q]$.

$$
\begin{aligned}
\hat{m}_t &= \frac{1}{2q+1} \sum_{j=-q}^{q} Y_{t-j} \\
&= \underbrace{\frac{1}{2q+1} \sum_{j=-q}^{q} m_{t-j}}_{\substack{\approx m_t \\ \text{for } q \text{ small}}} \quad + \quad \underbrace{\frac{1}{2q+1} \sum_{j=-q}^{q} W_{t-j}}_{\substack{\approx 0 \\ \text{for } q \text{ large}}}
\end{aligned}
$$

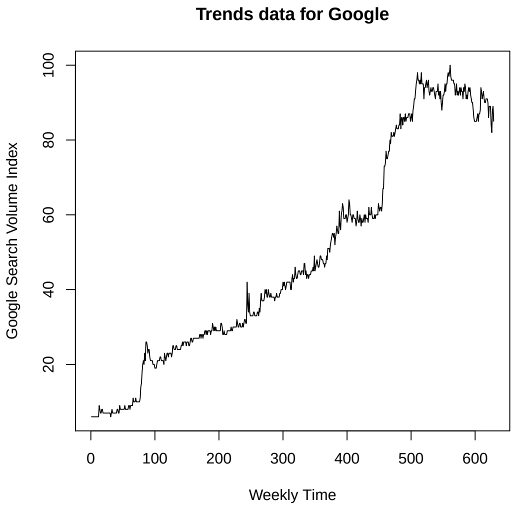
Bias-Variance Tradeoff!

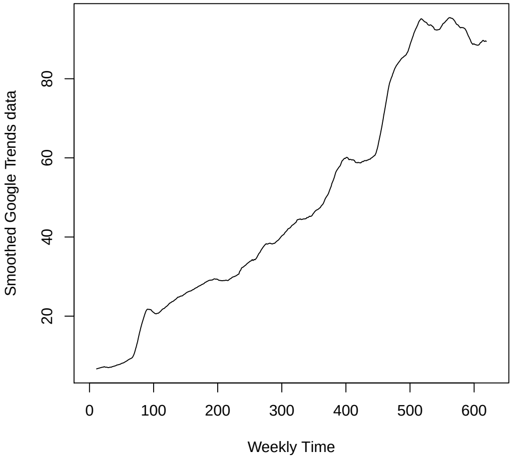
Bias-Variance Tradeoff $q = 5$



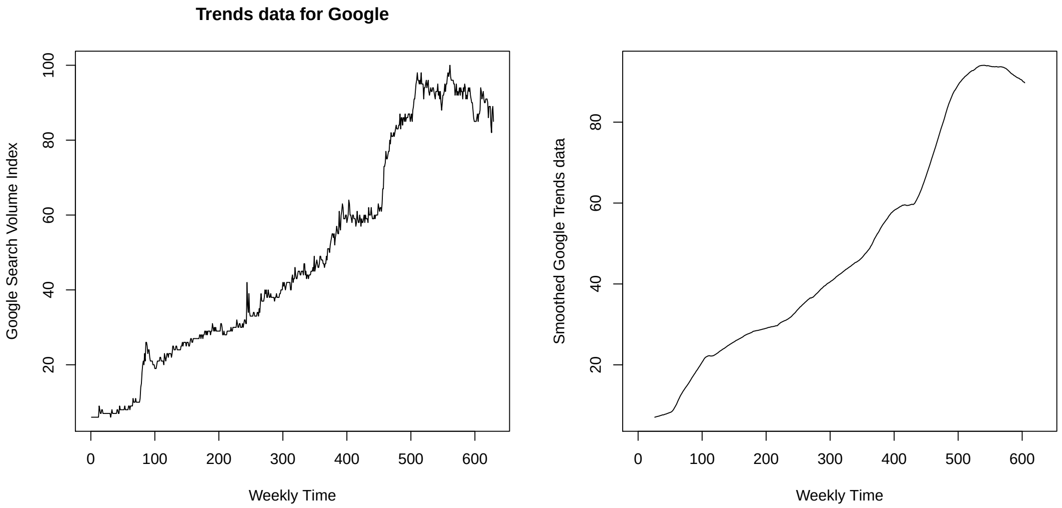


Bias-Variance Tradeoff $q = 10$

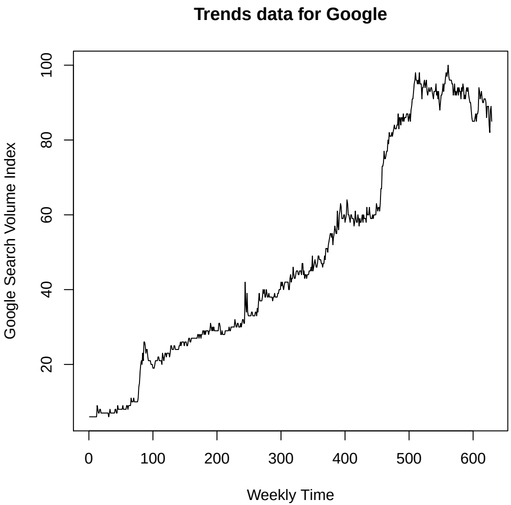


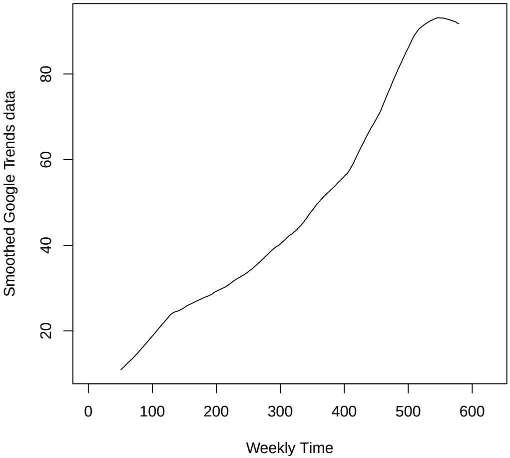


Bias-Variance Tradeoff $q = 25$

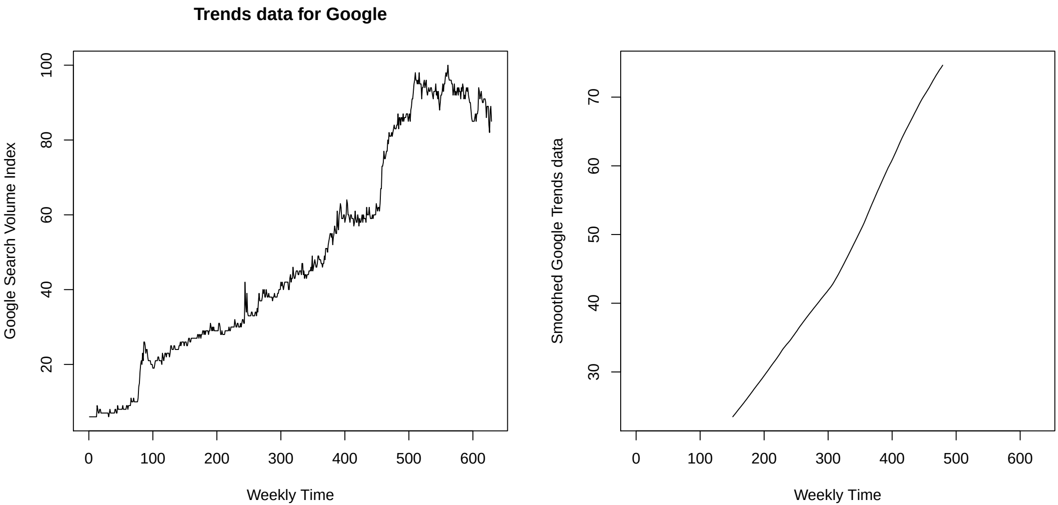


Bias-Variance Tradeoff $q = 50$

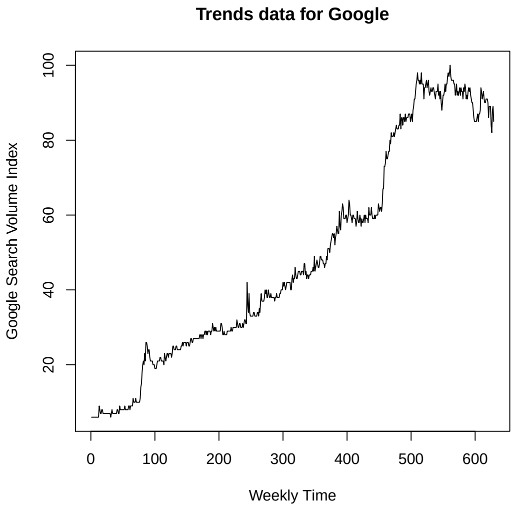


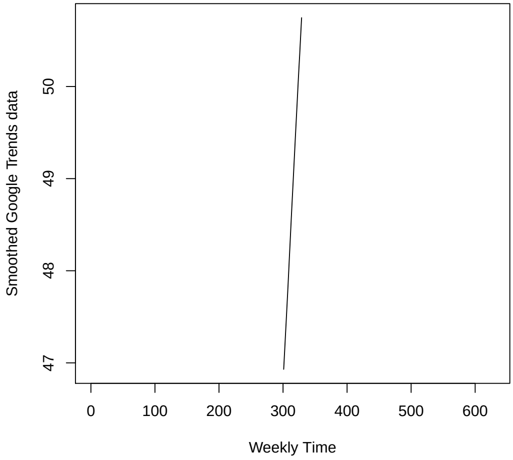


Bias-Variance Tradeoff $q = 150$



Bias-Variance Tradeoff $q = 300$





q-step Smoothing

Advantages

- No specific parametric form required (non-parametric)

Disadvantages:

- Selecting smoothing parameters such as size of neighborhood q is difficult
- No estimates for end-points
- No straight forward approach for predicting future values

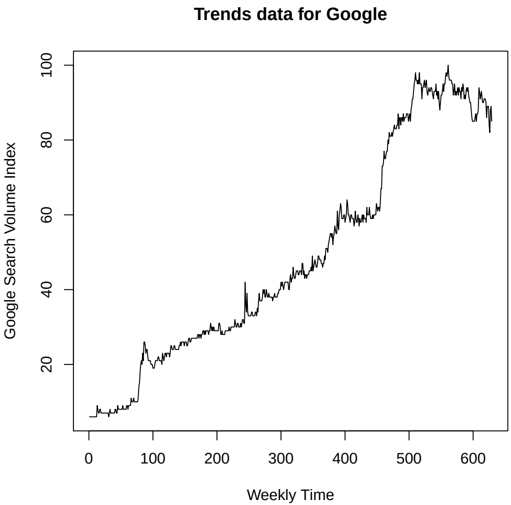
Unequal Weights

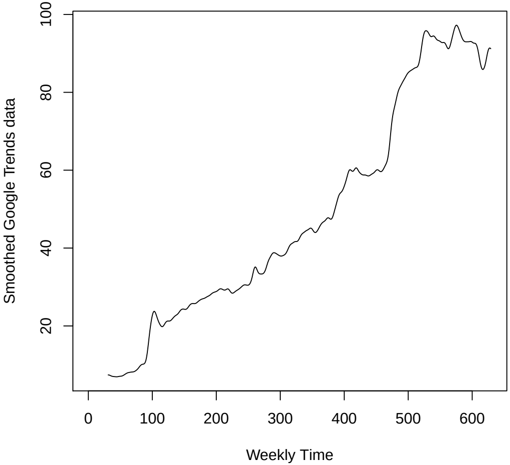
Binomial Weights

- q-step smoothing uses equal weights
- But it often makes sense to have weights decrease as distance in time (j) increases.
- We can do this with Binomial weights:

$$a_j = 2^{-q} \binom{q}{q/2 + j} \text{ for } = -q/2, -q/2 + 1, ..., -1, 0, 1, ..., q/2$$

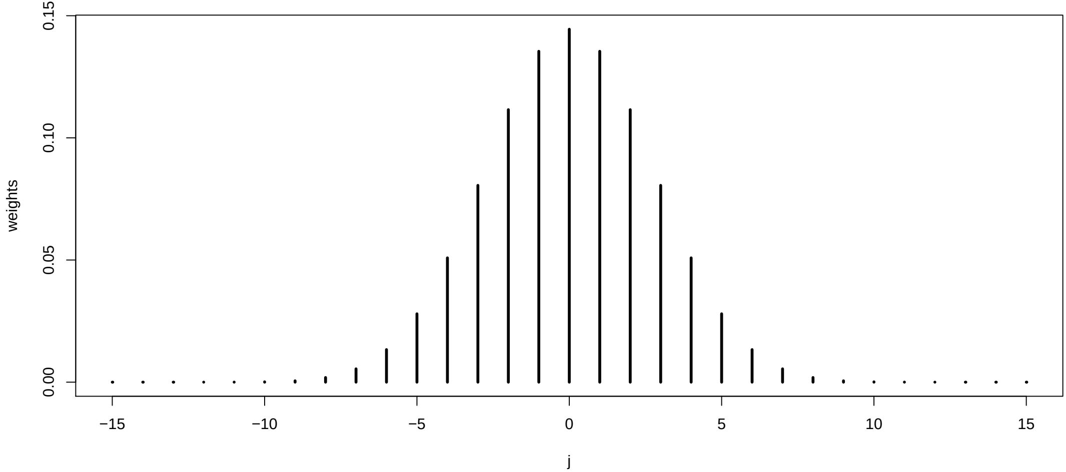
Binomial Weights





Binomial Weights

Note the weights:



Forecasting with Filters

- Note that the aforementioned filters cannot forecast directly as they use present/future values ($j \leq 0$), but if modified/rescaled to only use past values (j>0) they can fairly easily be used to forecast:

$$Y_t = a_1 Y_{t-1} + a_2 Y_{t-2} + ...$$

- The next filter is built like this.

Exponential Smoothing

- Exponential smoothing is designed to only use past values, and actually it uses all past observations

$$\hat{m}_t = c^*(\alpha Y_{t-1} + \alpha^2 Y_{t-2} + \alpha^3 Y_{t-3} \ldots)$$

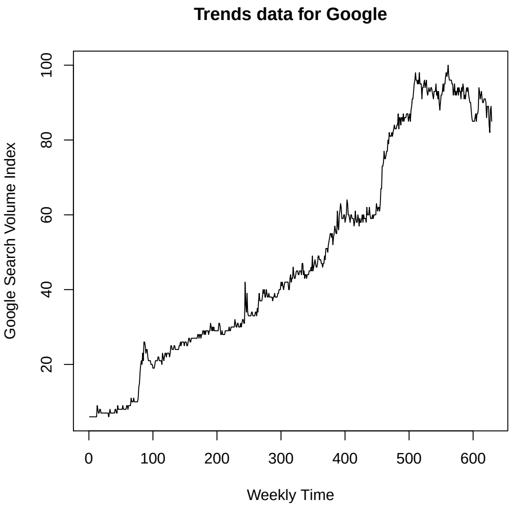
- Note that c^* is the constant needed for the coefficients to sum to one, such that $c^* = \frac{1-\alpha}{\alpha}$ and

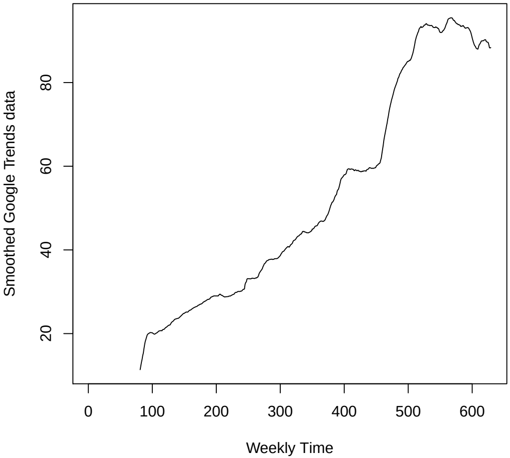
$$a_j = \alpha^{j-1}(1-\alpha)$$

  for $\alpha \in (0,1)$ and $j > 0$, $a_j = 0$ otherwise.

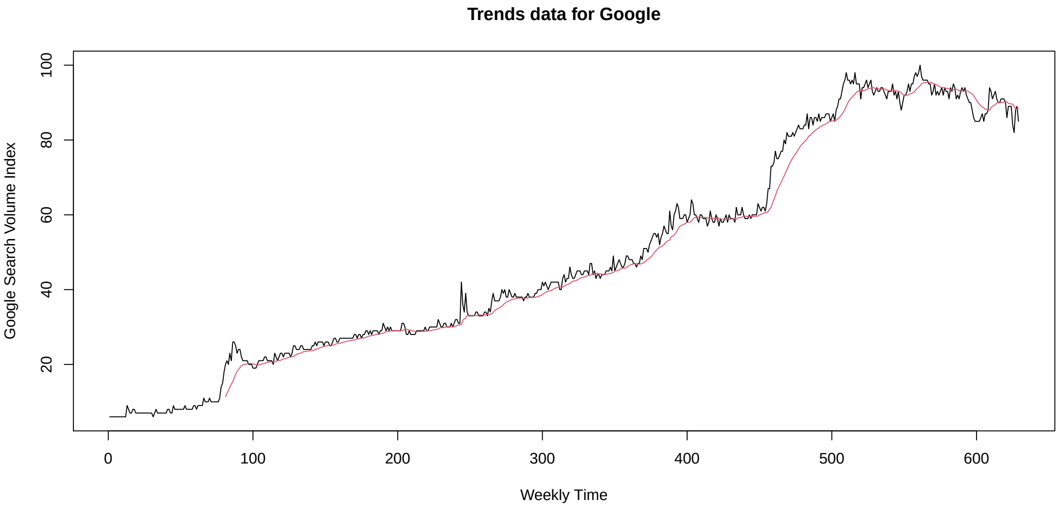
- Because we only use past values, we can forecast!

Exponential Smoothing





Careful!



Trends data for Google
Google Search Volume Index
Weekly Time
0
100
200
300
400
500
600
20
40
60
80
100

Preview of next time: Differencing

- Assume a linear trend $Y_t = a + bt + X_t$
- What if we set $a_0 = 1$ and $a_1 = -1$, 0 otherwise?
- Then the filtered series is $Y_t - Y_{t-1}$.
- What is the trend of the filtered series?