# Economic Time Series Filtering: An alternative approach with the neverhpfilter package

Justin M. Shea

June 1st, 2018

# Motivation

James Hamilton's working paper, **WHY YOU SHOULD NEVER USE THE HODRICK-PRESCOTT FILTER** (2017)<doi:10.3386/w23429> summarizes the problem with the popular filter in three points:

- (1) The HP filter produces series with spurious dynamic relations that have no basis in the underlying data-generating process.
- (2) Filtered values at the end of the sample are very different from those in the middle, and are also characterized by spurious dynamics.
- (3) A statistical formalization of the problem typically produces values for the smoothing parameter vastly at odds with common practice, e.g., a value for $\lambda$ far below $\mathbf{1600}$ for quarterly data.

# If you gave someone HP-Filtered data feeling like this. . .

# . . . In reality, its more like this:


( Inspired by the one and only Mara Averick @dataandme )

# The 4th point of Hamilton's abstract presents a solution

- (4) There's a better alternative. A regression of the variable at date $t + h$ on the four most recent values (for quarterly data) as of date $t$ offers a robust approach to detrending that achieves all the objectives sought by users of the HP filter with none of its drawbacks.

$$y_{t+8} = \beta_0 + \beta_1 y_t + \beta_2 y_{t-1} + \beta_3 y_{t-2} + \beta_4 y_{t-3} + v_{t+8}$$

Which can be rewritten as:

$$y_t = \beta_0 + \beta_1 y_{t-8} + \beta_2 y_{t-9} + \beta_3 y_{t-10} + \beta_4 y_{t-11} + v_t$$

# Do any of Hamilton's peers agree? From the cover page:

*I thank Daniel Leff for outstanding research assistance on this project and* ***Frank Diebold, Robert King, James Morley****, and anonymous referees for helpful comments on an earlier draft of this paper.*

# Implementing Hamilton's alternative: neverhpfilter package

**yth_glm**: fits a generalized linear model object of class **glm**.

$$y_{t+8} = \beta_0 + \beta_1 y_t + \beta_2 y_{t-1} + \beta_3 y_{t-2} + \beta_4 y_{t-3} + v_{t+8}$$

```
yth_glm(x, h = 8, p = 4, ...)
```

**yth_filter**: returns an **xts** object containing user defined combinations of the original, trend, cycle, and random walk series.

```
yth_filter(x, h = 8, p = 4,
           output = c("x", "trend", "cycle", "random"),
           ...)
```

In addition the package comes with **14** documented data sets used to reproduce the results of Hamilton(2017).

# Hamilton's alternative: Model estimation function

For model estimation, I settled on `glm` because. . . model object!

```
library(neverhpfilter)
gdp_model <- yth_glm(100*log(GDPC1), h = 8, p = 4)
```

| term | estimate | std.error | statistic | p.value |
|---|---|---|---|---|
| (Intercept) | 27.2025075 | 2.9638555 | 9.1780814 | 0.0000000 |
| xt_0 | 1.1722639 | 0.2336541 | 5.0170908 | 0.0000010 |
| xt_1 | -0.3432205 | 0.3858303 | -0.8895632 | 0.3745012 |
| xt_2 | -0.1296324 | 0.3856853 | -0.3361092 | 0.7370525 |
| xt_3 | 0.2769114 | 0.2320986 | 1.1930765 | 0.2338985 |

# Hamilton's alternative: Filtered series

```
library(neverhpfilter)
gdp_filtered <- yth_filter(100*log(GDPC1),
                           h = 8, p = 4,
                           output = c("x", "trend", "cycle"))

tail(gdp_filtered, 8)
```

```
##              GDPC1 GDPC1.trend GDPC1.cycle
## 2015 Q4 971.3998    971.0746  0.32512275
## 2016 Q1 971.5444    970.4246  1.11980556
## 2016 Q2 972.0977    971.9094  0.18831943
## 2016 Q3 972.7833    973.3109 -0.52760922
## 2016 Q4 973.2190    973.2501 -0.03104638
## 2017 Q1 973.5261    974.1597 -0.63363121
## 2017 Q2 974.2795    974.9659 -0.68630695
## 2017 Q3 975.0563    975.2427 -0.18635065
```

```
class(gdp_filtered)
```

# Got Dependencies?

**Dirk Eddelbuettel** Blog Code Publications Talks

**Wed, 28 Feb 2018**

## #17: Dependencies.

Dependencies are invitations for other people to break your package.
-- Josh Ulrich, private communication

# Got Dependencies?

**Dirk Eddelbuettel** Blog Code Publications Talks

**Wed, 28 Feb 2018**

**#17: Dependencies.**

> Dependencies are invitations for other people to break your package.
> -- Josh Ulrich, private communication

```
tools::package_dependencies("neverhpfilter")
```

```
## $neverhpfilter
## [1] "xts" "zoo"
```

# Why depend on xts?

- Safer
- Model functions accept and return xts objects of any periodicity.

```
class(GDPC1)
```

```
## [1] "xts" "zoo"
```

```
xts::periodicity(GDPC1)
```

```
## Quarterly periodicity from 1947 Q1 to 2017 Q3
```

```
y <- yth_filter(100*log(GDPC1), h = 8, p = 4)
class(y)
```

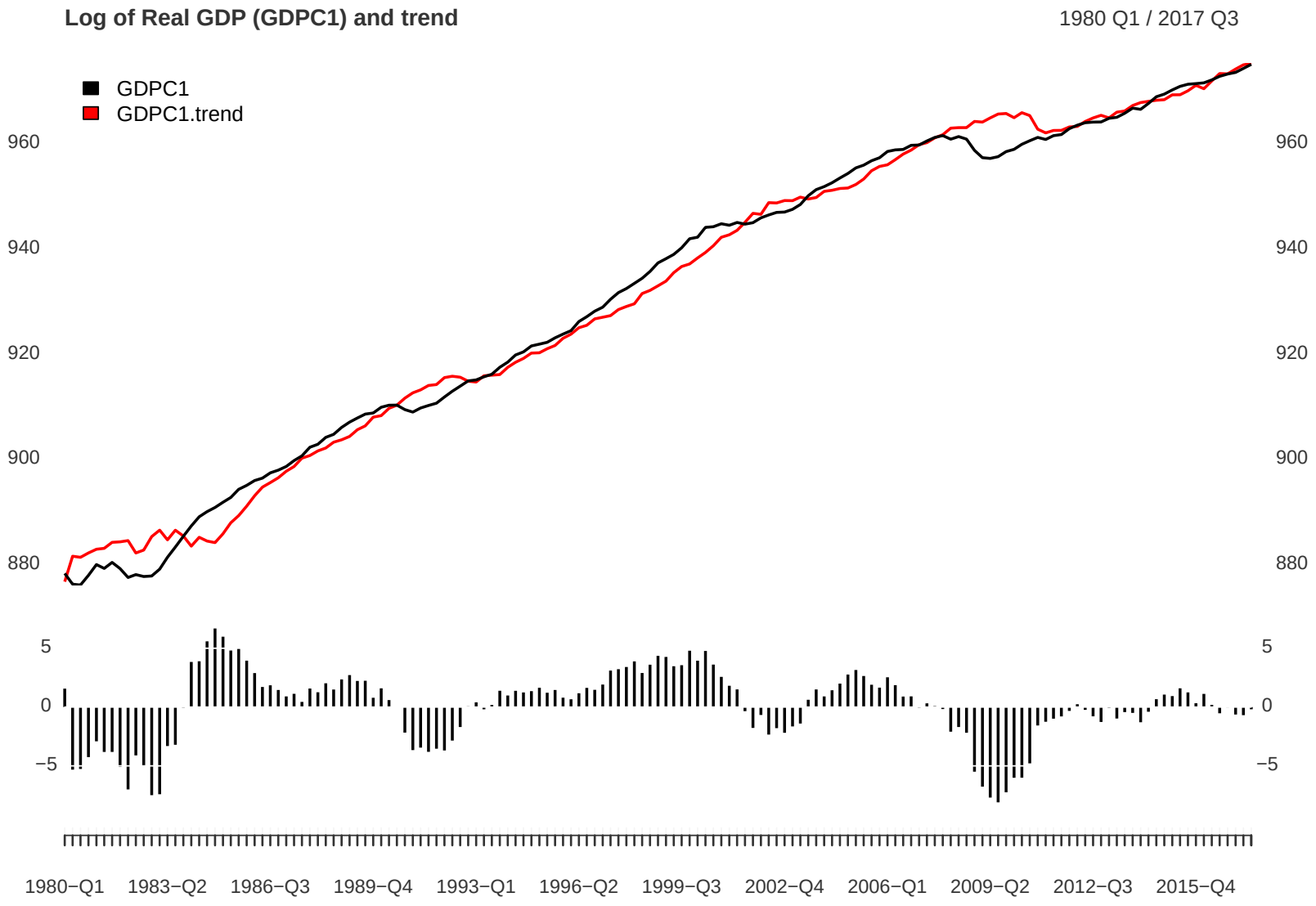
```
## [1] "xts" "zoo"
```

```
xts::periodicity(y)
```

```
## Quarterly periodicity from 1947 Q1 to 2017 Q3
```

# Why depend on xts?

`plot(x, ... )` quickly produces nice graphs.



# Reproducing Hamilton's solution

Hamilton's table 2 compared with estimates from neverhpfilter::yth_filter, sorted by standard deviation of the cycle component. yth_filter estimates are labeled with the suffix '.cycle'

| | cycle.sd | gdp.cor | random.sd | gdp.rand.cor | Sample |
|---|---|---|---|---|---|
| Unemployment-rate | 1.44 | -0.81 | 1.72 | -0.79 | 1948-1/2016-2 |
| UNRATENSA.cycle | 1.44 | -0.81 | 1.71 | -0.79 | 1948-Q1/2016-Q2 |
| 10-year-Treasury-yield | 1.46 | -0.05 | 1.51 | 0.08 | 1953-2/2016-2 |
| GS10.cycle | 1.46 | -0.05 | 1.51 | 0.08 | 1953-Q2/2016-Q2 |
| Fedfunds-rate | 2.78 | 0.33 | 3.03 | 0.40 | 1954-3/2016-2 |
| FEDFUNDS.cycle | 2.78 | 0.33 | 3.03 | 0.41 | 1954-Q3/2016-Q2 |
| Consumption | 2.85 | 0.79 | 3.04 | 0.82 | 1947-1/2016-1 |
| PCECC96.cycle | 2.86 | 0.79 | 3.04 | 0.82 | 1947-Q1/2016-Q1 |
| GDP-Deflator | 2.99 | 0.04 | 4.11 | -0.13 | 1947-1/2016-1 |
| GDPDEF.cycle | 2.99 | 0.03 | 4.10 | -0.13 | 1947-Q1/2016-Q1 |
| Employment | 3.09 | 0.85 | 3.32 | 0.85 | 1947-1/2016-2 |
| PAYEMS.cycle | 3.09 | 0.85 | 3.32 | 0.85 | 1947-Q1/2016-Q2 |
| GDP | 3.38 | 1.00 | 3.69 | 1.00 | 1947-1/2016-1 |

# James Hamilton is cool with open source R

Reply Reply All Forward

JH

Fri 1/26/2018 10:08 AM

James Hamilton <jhamilton@ucsd.edu>

**Re: "Why You Should Never Use the Hodrick-Prescott Filter" R package**

To Justin Shea

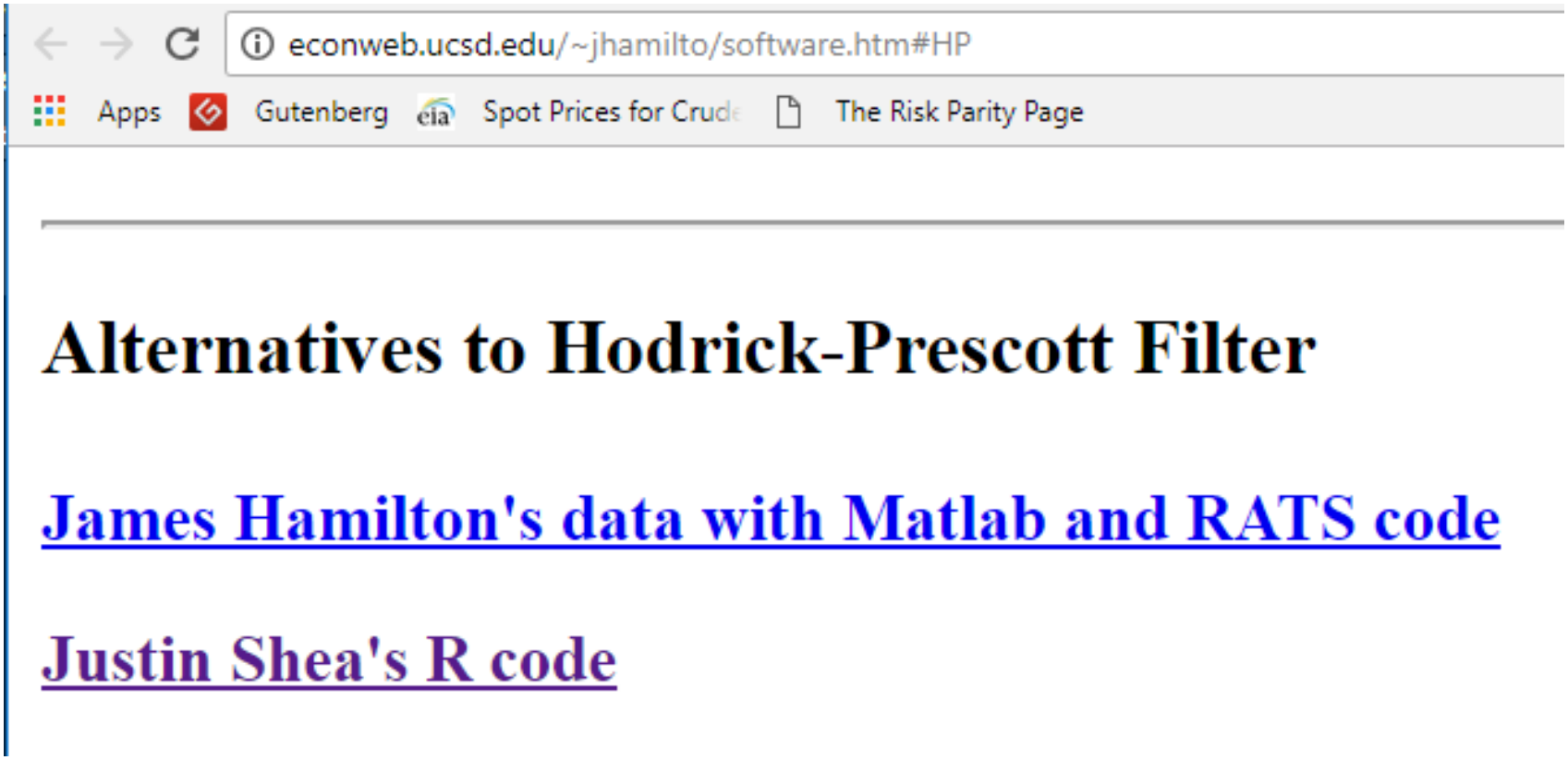
Bing Maps Action Items + Get more ac

Dear Justin,

Thanks for doing this!
I've linked to your page from both http://econweb.ucsd.edu/~jhamilto/ and http://econweb.ucsd.edu/~jhamilto/software.htm#HP

With gratitude,
Jim

# And he did it!


econweb.ucsd.edu/~jhamilto/software.htm#HP
Apps
Gutenberg
Spot Prices for Crud
The Risk Parity Page
Alternatives to Hodrick-Prescott Filter
James Hamilton's data with Matlab and RATS code
Justin Shea's R code

# Download the package and collaborate

On Cran:

```
install.packages("neverhpfilter")
```

Or dev version with current data:

```
devtools::install_github("JustinMShea/neverhpfilter")
```

# Thank you R/Finance!