Simulation study

Muhammad Yaseen

2026-11-05

Table of contents

1 Abstract **1**

2 The design **1**

3 Results **4**

4 Reading the results **6**

5 References **7**

1 Abstract

Hanif, Iqbal and Choudhary (2017) compare the HP, modified HP and fully modified HP filters on 30 simulated designs and report that the fully modified filter estimates the cycle better. This vignette repeats the comparison with `fmhp_simulation()` for quarterly series and for series averaged to annual data, with linear and nonlinear trends and both forms of the end-point weights. The results are precomputed by `tools/simulation-study.R` and shipped in `inst/extdata`; the vignette reads them, so it builds in seconds.

```
rmse  <- data.table::fread(system.file("extdata", "simulation_rmse.csv",
 ↪  package = "mhpfilter"))
power <- data.table::fread(system.file("extdata", "simulation_power.csv",
 ↪  package = "mhpfilter"))
n_rep <- 20
```

2 The design

Each series is a trend plus an AR(2) cycle (eqs. 10a and 10b of the paper). The 30 designs cross six cycles with five ratios $\sigma_\varepsilon/\sigma_\delta$ of the standard deviations of the trend and cycle shocks:

```r
mhp_designs(
    paper = "2017"
  )
```

```
    model sd_ratio ar1   ar2
 1:     1     10.0 0.9  0.01
 2:     2     10.0 1.2 -0.25
 3:     3     10.0 1.2 -0.40
 4:     4     10.0 1.2 -0.55
 5:     5     10.0 1.2 -0.75
 6:     6      5.0 0.9  0.01
 7:     7      5.0 1.2 -0.25
 8:     8      5.0 1.2 -0.40
 9:     9      5.0 1.2 -0.55
10:    10      5.0 1.2 -0.75
11:    11      2.0 0.9  0.01
12:    12      2.0 1.2 -0.25
13:    13      2.0 1.2 -0.40
14:    14      2.0 1.2 -0.55
15:    15      2.0 1.2 -0.75
16:    16      1.0 0.9  0.01
17:    17      1.0 1.2 -0.25
18:    18      1.0 1.2 -0.40
19:    19      1.0 1.2 -0.55
20:    20      1.0 1.2 -0.75
21:    21      0.5 0.9  0.01
22:    22      0.5 1.2 -0.25
23:    23      0.5 1.2 -0.40
24:    24      0.5 1.2 -0.55
25:    25      0.5 1.2 -0.75
26:    26     10.0 0.8  0.00
27:    27      5.0 0.8  0.00
28:    28      2.0 0.8  0.00
29:    29      1.0 0.8  0.00
30:    30      0.5 0.8  0.00
    model sd_ratio ar1   ar2
```

`mhp_simulate()` draws one series. With `trend = "nonlinear"` the drift of the trend varies smoothly over the sample; with `aggregate = "average"` quarterly values are averaged to annual ones and the HP filter uses $\lambda = 100$.

```r
s <-
  mhp_simulate(
      T         = 200
    , sd_ratio  = 1
    , ar        = c(1.2, -0.4)
    , sd_cycle  = 1
```

```
    , drift     = 0.5
    , trend     = "linear"
    , aggregate = "none"
    , seed      = 1
    )
s
```

```
       t           x       trend      cycle
  1:   1   1.679742  -0.1264538  1.8061955
  2:   2   1.037125   0.5571895  0.4799353
  3:   3   2.046342   0.2215609  1.8247815
  4:   4   3.930973   2.3168417  1.6141316
  5:   5   6.007540   3.1463495  2.8611906
 ---
196: 196 108.256252 104.6517201  3.6045314
197: 197 109.635655 106.5928778  3.0427775
198: 198 107.179656 106.0770304  1.1026259
199: 199 107.442699 106.9890051  0.4536938
200: 200 106.338047 107.1079290 -0.7698824
```

For each replicate the three filters estimate the cycle, and the root mean square error against the true cycle is taken over the whole series, the central 80% and the first and last 10%.

```
sim <-
  fmhp_simulation(
      designs    = mhp_designs(
          paper = c("2017", "2014")
        )[c(3, 13, 23)]
    , n_rep      = 5
    , T          = 200
    , trend      = "linear"
    , aggregate  = "none"
    , lambda     = NULL
    , max_lambda = 3000
    , seed       = 1
    , form       = "rows"
    )
sim$power
```

```
   model sd_ratio ar1  ar2 fmhp_better_hp fmhp_better_mhp
1:     3     10.0 1.2 -0.4            0.0             0.0
2:    13      2.0 1.2 -0.4            0.0             0.0
3:    23      0.5 1.2 -0.4            0.4             0.2
```

3 Results

The study ran 20 replicates of every design in each of eight settings. The share of replicates in which the fully modified filter has the smaller RMSE, averaged over the 30 designs:

```
power[, .(
    fmhp_better_hp  = round(x = mean(fmhp_better_hp), digits = 2)
  , fmhp_better_mhp = round(x = mean(fmhp_better_mhp), digits = 2)
  , designs_won_hp  = sum(fmhp_better_hp > 0.5)
  ), by = .(aggregate, trend, form)]
```

```
   aggregate     trend        form fmhp_better_hp fmhp_better_mhp
1:      none    linear        rows           0.20            0.11
2:      none    linear differences           0.25            0.17
3:      none nonlinear        rows           0.16            0.10
4:      none nonlinear differences           0.21            0.14
5:   average    linear        rows           0.06            0.05
6:   average    linear differences           0.07            0.07
7:   average nonlinear        rows           0.01            0.01
8:   average nonlinear differences           0.03            0.03
   designs_won_hp
1:              0
2:              3
3:              0
4:              1
5:              2
6:              1
7:              0
8:              0
```

By the noise ratio (quarterly series, linear trend, the form of the MATLAB code):

```
power[aggregate == "none" & trend == "linear" & form == "rows",
      .(
          fmhp_better_hp  = round(x = mean(fmhp_better_hp), digits = 2)
        , fmhp_better_mhp = round(x = mean(fmhp_better_mhp), digits = 2)
        ),
      by = sd_ratio]
```

```
   sd_ratio fmhp_better_hp fmhp_better_mhp
1:     10.0           0.22            0.00
2:      5.0           0.22            0.00
3:      2.0           0.09            0.03
4:      1.0           0.14            0.19
5:      0.5           0.33            0.34
```

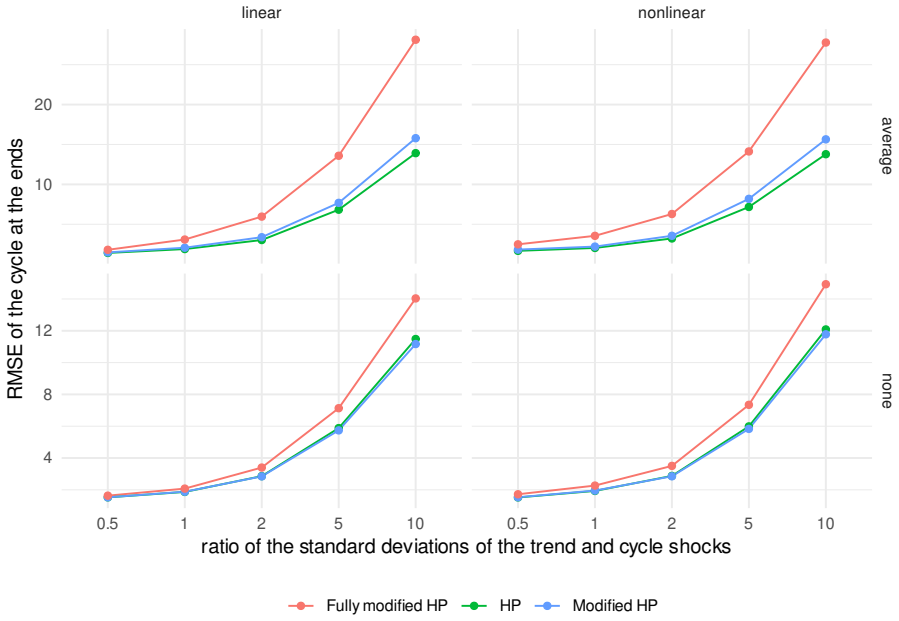
and the average RMSE of each filter, over the whole series, the middle and the

ends:

```
rmse[aggregate == "none" & trend == "linear" & form == "rows",
     .(
         rmse   = round(x = mean(rmse), digits = 3)
       , middle = round(x = mean(rmse_middle), digits = 3)
       , ends   = round(x = mean(rmse_ends), digits = 3)
       ),
     by = method]
```

```
              method  rmse middle  ends
1:                HP 5.122  5.188 4.723
2:       Modified HP 4.989  5.047 4.632
3: Fully modified HP 5.294  5.162 5.655
```

```
d <- rmse[form == "rows", .(rmse_ends = mean(rmse_ends)), by =
 ↪  .(aggregate, trend, method, sd_ratio)]
ggplot2::ggplot(
    data    = d
  , mapping = ggplot2::aes(x = factor(sd_ratio), y = rmse_ends, colour =
 ↪  method, group = method)
  ) +
  ggplot2::geom_line() +
  ggplot2::geom_point() +
  ggplot2::facet_grid(
      rows   = ggplot2::vars(aggregate)
    , cols   = ggplot2::vars(trend)
    , scales = "free_y"
    ) +
  ggplot2::labs(x = "ratio of the standard deviations of the trend and
 ↪  cycle shocks",
                y = "RMSE of the cycle at the ends", colour = NULL) +
  ggplot2::theme_minimal() +
  ggplot2::theme(legend.position = "bottom")
```



4 Reading the results

The tables above are the evidence; the points below only summarise them.

- The paper reports (Table 1) that the fully modified filter beats the HP filter in 100% of the replicates of every design and every setting. Here it has the smaller RMSE in 20% of the replicates for quarterly series with a linear trend and in 6% for series averaged to annual data, and it beats the HP filter in more than half of the replicates in 7 of the 240 combinations of design and setting.
- The difference is largest at the ends, where the method is meant to help: for quarterly series with a linear trend the RMSE at the ends is 4.72 for the HP filter and 5.66 for the fully modified filter, against 5.19 and 5.16 in the middle.
- The loss that chooses the end-point weights measures how far the gain at each date is from the gain in the middle; it does not depend on the data and does not measure how close the estimated cycle is to the true one. A likely reason for the result: raising λ near the ends makes the trend stiffer there, so the estimated cycle takes in more of the trend shocks.
- The filter used here is the one of the authors' MATLAB code (the tests check it to 10^{-8}), so the difference does not come from the implementation. It can come from the design: the paper does not fully report how its trends were drawn, the length of its series or how its RMSE was computed, so these results are not a reproduction of its Tables 1 and 2. They are 20

replicates per design; the shares are rough to a few percentage points.
- The two forms of the end-point weights give similar results.

To rerun the study with more replicates:

```
# from the package folder; 100 replicates take about three hours
system2(command = "Rscript", args = c("tools/simulation-study.R", "100"))
```

5 References

Choudhary, M. A., Hanif, M. N. and Iqbal, J. (2014). On smoothing macroeconomic time series using the modified HP filter. *Applied Economics*, 46(19), 2205–2214.

Hanif, M. N., Iqbal, J. and Choudhary, M. A. (2017). Fully modified HP filter. *SBP Working Paper Series*, No. 88, State Bank of Pakistan.