replication

Zhirui Wang

August 5, 2016

table1

```
Expected.Inflation <- read.csv("C:/Users/wang_/Desktop/Expected Inflation.csv")
CPIAUCSL <- read.csv("C:/Users/wang_/Desktop/CPIAUCSL.csv")
five_five_year <- 2*Expected.Inflation[,11]-Expected.Inflation[,6]
real <- CPIAUCSL[,2]/100
one_year <- Expected.Inflation[,2]
f_five_five_year <- five_five_year[-(1:12)]
d_five_five_year <- f_five_five_year-five_five_year[1:403]
x <- real[1:403]-one_year[1:403]
summary(lm(d_five_five_year[1:300]~x[1:300]))
```

```
##
## Call:
## lm(formula = d_five_five_year[1:300] ~ x[1:300])
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -0.0083737 -0.0025571 -0.0001024  0.0023564  0.0098137
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.0005346  0.0002172  -2.461   0.0144 *
## x[1:300]     0.1955986  0.0175912  11.119   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.003643 on 298 degrees of freedom
## Multiple R-squared:  0.2932, Adjusted R-squared:  0.2909
## F-statistic: 123.6 on 1 and 298 DF,  p-value: < 2.2e-16
```

```
summary(lm(d_five_five_year[1:84]~x[1:84]))
```

```
##
## Call:
## lm(formula = d_five_five_year[1:84] ~ x[1:84])
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -0.0067882 -0.0033685 -0.0000634  0.0024868  0.0094249
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 0.0004881  0.0005081   0.961     0.34
```

```
## x[1:84]     0.2928575  0.0294328   9.950 9.37e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.003957 on 82 degrees of freedom
## Multiple R-squared:  0.547,  Adjusted R-squared:  0.5414
## F-statistic:    99 on 1 and 82 DF,  p-value: 9.37e-16
```

```
summary(lm(d_five_five_year[85:204]~x[85:204]))
```

```
##
## Call:
## lm(formula = d_five_five_year[85:204] ~ x[85:204])
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -0.0078915 -0.0024553  0.0000486  0.0018900  0.0085799
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.0001988  0.0003461  -0.574    0.567
## x[85:204]    0.1714095  0.0370038   4.632 9.39e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.003473 on 118 degrees of freedom
## Multiple R-squared:  0.1539, Adjusted R-squared:  0.1467
## F-statistic: 21.46 on 1 and 118 DF,  p-value: 9.385e-06
```

```
summary(lm(d_five_five_year[205:300]~x[205:300]))
```

```
##
## Call:
## lm(formula = d_five_five_year[205:300] ~ x[205:300])
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -0.0071721 -0.0016671  0.0002234  0.0017982  0.0068163
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.0008468  0.0003277  -2.584   0.0113 *
## x[205:300]   0.0821827  0.0317788   2.586   0.0112 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.003071 on 94 degrees of freedom
## Multiple R-squared:  0.06642,    Adjusted R-squared:  0.05649
## F-statistic: 6.688 on 1 and 94 DF,  p-value: 0.01124
```

```
summary(lm(d_five_five_year[301:402]~x[301:402]))
```

```
##
## Call:
## lm(formula = d_five_five_year[301:402] ~ x[301:402])
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -0.0041372 -0.0012659 -0.0000061  0.0012909  0.0046577
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.0006451  0.0002131  -3.027  0.00314 **
## x[301:402]   0.0208158  0.0129230   1.611  0.11039
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.002151 on 100 degrees of freedom
## Multiple R-squared:  0.02529,    Adjusted R-squared:  0.01554
## F-statistic: 2.595 on 1 and 100 DF,  p-value: 0.1104
```

table2

```
library(car)
library(lmtest)
```

```
## Loading required package: zoo
```

```
##
## Attaching package: 'zoo'
```

```
## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

```
CPIAUCSL2 <- read.csv("C:/Users/wang_/Desktop/CPIAUCSL (3).csv")
INDPRO <- read.csv("C:/Users/wang_/Desktop/INDPRO.csv")
pi <- CPIAUCSL2[,2,drop=F]
y <- INDPRO[,2,drop=F]
##US result 83-89
depen <- pi[432:515,]
indust <- y[1:84,]
indepen <- NA
indepen <- pi[431:514,,drop=F]
for (i in 1:47) {
  indepen <- cbind(indepen,pi[(431-i):(514-i),,drop=F])
}
(formula=depen ~ alpha+indepen[,1]*gamma+indepen[,2]*gamma*(1-gamma)+indepen[,3]*gamma*(1-gamma)^2
+indepen[,4]*gamma*(1-gamma)^3+indepen[,5]*gamma*(1-gamma)^4+indepen[,6]*gamma*(1-gamma)^5
+indepen[,7]*gamma*(1-gamma)^6+indepen[,8]*gamma*(1-gamma)^7+indepen[,9]*gamma*(1-gamma)^8
+indepen[,10]*gamma*(1-gamma)^9+indepen[,11]*gamma*(1-gamma)^10+indepen[,12]*gamma*(1-gamma)^11
+indepen[,13]*gamma*(1-gamma)^12+indepen[,14]*gamma*(1-gamma)^13+indepen[,15]*gamma*(1-gamma)^14
```

```
+indepen[,16]*gamma*(1-gamma)^15+indepen[,17]*gamma*(1-gamma)^16+indepen[,18]*gamma*(1-gamma)^17
+indepen[,19]*gamma*(1-gamma)^18+indepen[,20]*gamma*(1-gamma)^19+indepen[,21]*gamma*(1-gamma)^20
+indepen[,22]*gamma*(1-gamma)^21+indepen[,23]*gamma*(1-gamma)^22+indepen[,24]*gamma*(1-gamma)^23
+indepen[,25]*gamma*(1-gamma)^24+indepen[,26]*gamma*(1-gamma)^25+indepen[,27]*gamma*(1-gamma)^26
+indepen[,28]*gamma*(1-gamma)^27+indepen[,29]*gamma*(1-gamma)^28+indepen[,30]*gamma*(1-gamma)^29
+indepen[,31]*gamma*(1-gamma)^30+indepen[,32]*gamma*(1-gamma)^31+indepen[,33]*gamma*(1-gamma)^32
+indepen[,34]*gamma*(1-gamma)^33+indepen[,35]*gamma*(1-gamma)^34+indepen[,36]*gamma*(1-gamma)^35
+indepen[,37]*gamma*(1-gamma)^36+indepen[,38]*gamma*(1-gamma)^37+indepen[,39]*gamma*(1-gamma)^38
+indepen[,40]*gamma*(1-gamma)^39+indepen[,41]*gamma*(1-gamma)^40+indepen[,42]*gamma*(1-gamma)^41
+indepen[,43]*gamma*(1-gamma)^42+indepen[,44]*gamma*(1-gamma)^43+indepen[,45]*gamma*(1-gamma)^44
+indepen[,46]*gamma*(1-gamma)^45+indepen[,47]*gamma*(1-gamma)^46+indepen[,48]*gamma*(1-gamma)^47
+lamda*indust)
```

```
## depen ~ alpha + indepen[, 1] * gamma + indepen[, 2] * gamma *
##     (1 - gamma) + indepen[, 3] * gamma * (1 - gamma)^2 + indepen[,
##     4] * gamma * (1 - gamma)^3 + indepen[, 5] * gamma * (1 -
##     gamma)^4 + indepen[, 6] * gamma * (1 - gamma)^5 + indepen[,
##     7] * gamma * (1 - gamma)^6 + indepen[, 8] * gamma * (1 -
##     gamma)^7 + indepen[, 9] * gamma * (1 - gamma)^8 + indepen[,
##     10] * gamma * (1 - gamma)^9 + indepen[, 11] * gamma * (1 -
##     gamma)^10 + indepen[, 12] * gamma * (1 - gamma)^11 + indepen[,
##     13] * gamma * (1 - gamma)^12 + indepen[, 14] * gamma * (1 -
##     gamma)^13 + indepen[, 15] * gamma * (1 - gamma)^14 + indepen[,
##     16] * gamma * (1 - gamma)^15 + indepen[, 17] * gamma * (1 -
##     gamma)^16 + indepen[, 18] * gamma * (1 - gamma)^17 + indepen[,
##     19] * gamma * (1 - gamma)^18 + indepen[, 20] * gamma * (1 -
##     gamma)^19 + indepen[, 21] * gamma * (1 - gamma)^20 + indepen[,
##     22] * gamma * (1 - gamma)^21 + indepen[, 23] * gamma * (1 -
##     gamma)^22 + indepen[, 24] * gamma * (1 - gamma)^23 + indepen[,
##     25] * gamma * (1 - gamma)^24 + indepen[, 26] * gamma * (1 -
##     gamma)^25 + indepen[, 27] * gamma * (1 - gamma)^26 + indepen[,
##     28] * gamma * (1 - gamma)^27 + indepen[, 29] * gamma * (1 -
##     gamma)^28 + indepen[, 30] * gamma * (1 - gamma)^29 + indepen[,
##     31] * gamma * (1 - gamma)^30 + indepen[, 32] * gamma * (1 -
##     gamma)^31 + indepen[, 33] * gamma * (1 - gamma)^32 + indepen[,
##     34] * gamma * (1 - gamma)^33 + indepen[, 35] * gamma * (1 -
##     gamma)^34 + indepen[, 36] * gamma * (1 - gamma)^35 + indepen[,
##     37] * gamma * (1 - gamma)^36 + indepen[, 38] * gamma * (1 -
##     gamma)^37 + indepen[, 39] * gamma * (1 - gamma)^38 + indepen[,
##     40] * gamma * (1 - gamma)^39 + indepen[, 41] * gamma * (1 -
##     gamma)^40 + indepen[, 42] * gamma * (1 - gamma)^41 + indepen[,
##     43] * gamma * (1 - gamma)^42 + indepen[, 44] * gamma * (1 -
##     gamma)^43 + indepen[, 45] * gamma * (1 - gamma)^44 + indepen[,
##     46] * gamma * (1 - gamma)^45 + indepen[, 47] * gamma * (1 -
##     gamma)^46 + indepen[, 48] * gamma * (1 - gamma)^47 + lamda *
##     indust
```

```
m1 <- nls(formula,start=list(alpha = 0, gamma = 0.246, lamda = 0))
coeftest(m1)
```

```
##
## t test of coefficients:
##
```

```
##        Estimate Std. Error t value  Pr(>|t|)
## alpha -0.292986   0.283880 -1.0321 0.3051092
## gamma  0.160286   0.042962  3.7309 0.0003528 ***
## lamda  0.095828   0.036529  2.6234 0.0104008 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##US result 90-99
depen <- pi[516:635,]
indust <- y[85:204,]
indepen <- NA
indepen <- pi[515:634,,drop=F]
for (i in 1:47) {
  indepen <- cbind(indepen,pi[(515-i):(634-i),,drop=F])
}
m2 <- nls(formula,start=list(alpha = 0, gamma = 0.088, lamda = 0))
coeftest(m2)
```

```
##
## t test of coefficients:
##
##        Estimate Std. Error t value Pr(>|t|)
## alpha -0.179911   0.196866 -0.9139  0.36266
## gamma  0.118297   0.046155  2.5631  0.01164 *
## lamda  0.017529   0.026685  0.6569  0.51255
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##US result 00-07
depen <- pi[636:731,]
indust <- y[205:300,]
indepen <- NA
indepen <- pi[635:730,,drop=F]
for (i in 1:399) {
  indepen <- cbind(indepen,pi[(635-i):(730-i),,drop=F])
}
m3 <- nls(formula,start=list(alpha = 0, gamma = 0.05, lamda = 0))
coeftest(m3)
```

```
##
## t test of coefficients:
##
##        Estimate Std. Error t value Pr(>|t|)
## alpha  5.059327   3.122018  1.6205   0.1085
## gamma -0.013135   0.014140 -0.9289   0.3553
## lamda -0.049008   0.056730 -0.8639   0.3899
```

```
##US result 08-16
depen <- pi[732:833,]
indust <- y[301:402,]
indepen <- NA
indepen <- pi[731:832,,drop=F]
for (i in 1:47) {
```

```
  indepen <- cbind(indepen,pi[(731-i):(832-i),,drop=F])
}
m4 <- nls(formula,start=list(alpha = 0, gamma = 0, lamda = 0))
coeftest(m4)
```

```
##
## t test of coefficients:
##
##         Estimate Std. Error t value Pr(>|t|)
## alpha  1.9266156  1.4482888  1.3303   0.1865
## gamma -0.0039609  0.0116430 -0.3402   0.7344
## lamda  0.0546951  0.0428609  1.2761   0.2049
```

table3

```
library(car)
library(lmtest)
CPI_headline <- read.csv("D:/Dropbox/16summer/Macroeconometrics/Paper/CPI_headline.csv")
Industrial.Production.Index <- read.csv("D:/Dropbox/16summer/Macroeconometrics/Paper/Industrial Producti
a <- as.character(unique(Industrial.Production.Index[,1]))
table <- matrix(rep(NA,164),41,4)
rownames(table) <- a
colnames(table) <- c("Estimate", "Std. Error", "t value","Pr(>|t|)")
for(j in c(1:41)){
  b <- a[j]
  pi <- CPI_headline[CPI_headline[,1]==b,7,drop=F]
  pi1 <- data.frame(rep(NA,270))
  for (i in 1:270) {
    pi1[i,1] <- (pi[i+1,1,drop=F]-pi[i,1,drop=F])/pi[i,1,drop=F]
  }
  pi<- ((1+pi1[,1,drop=F])^4-1)*100
  y <- Industrial.Production.Index[Industrial.Production.Index[,1]==b,7,drop=F]
  y1 <- rep(NA,216)
  for (i in 1:216) {
    y1[i] <- (y[i+1,1]-y[i,1])/y[i,1]
  }
  y <- ((1+y1)^4-1)*100
  depen <- pi[25:240,1]
  indust <- y[1:216]
  indepen <- NA
  indepen <- pi[24:239,1,drop=F]
  for (i in 1:23) {
    indepen <- cbind(indepen,pi[(24-i):(239-i),1,drop=F])
  }
  (formula=depen ~ alpha+indepen[,1]*gamma+indepen[,2]*gamma*(1-gamma)+indepen[,3]*gamma*(1-gamma)^2
  +indepen[,4]*gamma*(1-gamma)^3+indepen[,5]*gamma*(1-gamma)^4+indepen[,6]*gamma*(1-gamma)^5
  +indepen[,7]*gamma*(1-gamma)^6+indepen[,8]*gamma*(1-gamma)^7+indepen[,9]*gamma*(1-gamma)^8
  +indepen[,10]*gamma*(1-gamma)^9+indepen[,11]*gamma*(1-gamma)^10+indepen[,12]*gamma*(1-gamma)^11
  +indepen[,13]*gamma*(1-gamma)^12+indepen[,14]*gamma*(1-gamma)^13+indepen[,15]*gamma*(1-gamma)^14
  +indepen[,16]*gamma*(1-gamma)^15+indepen[,17]*gamma*(1-gamma)^16+indepen[,18]*gamma*(1-gamma)^17
```

```
  +indepen[,19]*gamma*(1-gamma)^18+indepen[,20]*gamma*(1-gamma)^19+indepen[,21]*gamma*(1-gamma)^20
  +indepen[,22]*gamma*(1-gamma)^21+indepen[,23]*gamma*(1-gamma)^22+indepen[,24]*gamma*(1-gamma)^23
  +lamda*indust)
  m <- nls(formula,start=list(alpha = 0, gamma = 0.2, lamda = 0),control = list(maxiter=500))
  table[j,] <- coeftest(m)[2,]
}
table[order(table[,1],decreasing = TRUE),]
```

```
##            Estimate  Std. Error      t value     Pr(>|t|)
## COL    1.0668377809 0.069037284 15.453067020 1.244076e-36
## BRA    0.7729898557 0.066918317 11.551244640 2.782680e-24
## MEX    0.7694215213 0.065657152 11.718776988 8.378091e-25
## TUR    0.3181680392 0.050059851  6.355752772 1.240881e-09
## CHL    0.2769956313 0.046680740  5.933831258 1.185653e-08
## LVA    0.2282855762 0.045963856  4.966632362 1.495137e-06
## EST    0.2263836677 0.045871429  4.935177965 1.719336e-06
## ITA    0.1737614863 0.039043613  4.450445884 1.381221e-05
## RUS    0.1733319503 0.038911596  4.454506356 1.357532e-05
## ISL    0.1317081252 0.033699305  3.908333627 1.248607e-04
## IRL    0.1115373441 0.031846263  3.502368373 5.617800e-04
## OECDE  0.0650745957 0.018300709  3.555851149 4.641750e-04
## HUN    0.0602019979 0.016203428  3.715386462 2.592472e-04
## POL    0.0571580946 0.012967485  4.407801209 1.655234e-05
## SVN    0.0473823481 0.018830130  2.516304875 1.259654e-02
## ISR    0.0472184338 0.017344520  2.722383444 7.018166e-03
## CZE    0.0455461289 0.020116382  2.264131213 2.457311e-02
## OECD   0.0433236392 0.017563543  2.466679933 1.442647e-02
## SVK    0.0325104376 0.019451892  1.671325245 9.612596e-02
## EU28   0.0254939871 0.018853827  1.352191651 1.777479e-01
## IND    0.0249795358 0.018334527  1.362431455 1.745006e-01
## GBR    0.0179175253 0.021715631  0.825098093 4.102390e-01
## FIN    0.0142126982 0.016273440  0.873367765 3.834462e-01
## KOR    0.0139818737 0.019140844  0.730473223 4.659037e-01
## EA19   0.0128877605 0.027224181  0.473393880 6.364170e-01
## PRT    0.0112982013 0.019652500  0.574898936 5.659664e-01
## JPN    0.0069948025 0.020406023  0.342781274 7.321011e-01
## GRC    0.0061565199 0.027087539  0.227282364 8.204222e-01
## NLD    0.0035406291 0.026646181  0.132875666 8.944172e-01
## SWE    0.0026206338 0.017523478  0.149549867 8.812613e-01
## ESP    0.0013359781 0.020890927  0.063950158 9.490699e-01
## AUT    0.0000349162 0.016643455  0.002097894 9.983281e-01
## DNK   -0.0016101618 0.029531377 -0.054523764 9.565690e-01
## LUX   -0.0027055734 0.030392819 -0.089020154 9.291495e-01
## FRA   -0.0032410723 0.018162256 -0.178450979 8.585384e-01
## G-7   -0.0039792518 0.016278685 -0.244445528 8.071211e-01
## USA   -0.0053446759 0.015597976 -0.342651892 7.321983e-01
## NOR   -0.0061274729 0.020463880 -0.299428695 7.649049e-01
## DEU   -0.0091392589 0.019220595 -0.475493019 6.349229e-01
## CAN   -0.0091584981 0.017093189 -0.535798095 5.926571e-01
## BEL   -0.0157700711 0.009942748 -1.586087758 1.142028e-01
```

table4

```
library(car)
library(lmtest)
CPI_core <- read.csv("D:/Dropbox/16summer/Macroeconometrics/Paper/CPI_core.csv")
Industrial.Production.Index <- read.csv("D:/Dropbox/16summer/Macroeconometrics/Paper/Industrial Producti
a <- intersect(unique(CPI_core[,1]),unique(Industrial.Production.Index[,1]))
table <- matrix(rep(NA,1152),38,4)
```

```
## Warning in matrix(rep(NA, 1152), 38, 4): data length [1152] is not a sub-
## multiple or multiple of the number of rows [38]
```

```
rownames(table) <- a
colnames(table) <- c("Estimate", "Std. Error", "t value","Pr(>|t|)")
for(j in c(1:38)){
  b <- a[j]
  pi <- CPI_core[CPI_core[,1]==b,7,drop=F]
  pi1 <- data.frame(rep(NA,270))
  for (i in 1:270) {
    pi1[i,1] <- (pi[i+1,1,drop=F]-pi[i,1,drop=F])/pi[i,1,drop=F]
  }
  pi<- ((1+pi1[,1,drop=F])^4-1)*100
  y <- Industrial.Production.Index[Industrial.Production.Index[,1]==b,7,drop=F]
  y1 <- rep(NA,216)
  for (i in 1:216) {
    y1[i] <- (y[i+1,1]-y[i,1])/y[i,1]
  }
  y <- ((1+y1)^4-1)*100
  depen <- pi[25:240,1]
  indust <- y[1:216]
  indepen <- NA
  indepen <- pi[24:239,1,drop=F]
  for (i in 1:23) {
    indepen <- cbind(indepen,pi[(24-i):(239-i),1,drop=F])
  }
  (formula=depen ~ alpha+indepen[,1]*gamma+indepen[,2]*gamma*(1-gamma)+indepen[,3]*gamma*(1-gamma)^2
  +indepen[,4]*gamma*(1-gamma)^3+indepen[,5]*gamma*(1-gamma)^4+indepen[,6]*gamma*(1-gamma)^5
  +indepen[,7]*gamma*(1-gamma)^6+indepen[,8]*gamma*(1-gamma)^7+indepen[,9]*gamma*(1-gamma)^8
  +indepen[,10]*gamma*(1-gamma)^9+indepen[,11]*gamma*(1-gamma)^10+indepen[,12]*gamma*(1-gamma)^11
  +indepen[,13]*gamma*(1-gamma)^12+indepen[,14]*gamma*(1-gamma)^13+indepen[,15]*gamma*(1-gamma)^14
  +indepen[,16]*gamma*(1-gamma)^15+indepen[,17]*gamma*(1-gamma)^16+indepen[,18]*gamma*(1-gamma)^17
  +indepen[,19]*gamma*(1-gamma)^18+indepen[,20]*gamma*(1-gamma)^19+indepen[,21]*gamma*(1-gamma)^20
  +indepen[,22]*gamma*(1-gamma)^21+indepen[,23]*gamma*(1-gamma)^22+indepen[,24]*gamma*(1-gamma)^23
  +lamda*indust)
  m <- nls(formula,start=list(alpha = 0, gamma = 0.5, lamda = 0),control = list(maxiter=5000))
  table[j,] <- coeftest(m)[2,]
}
table[order(table[,1],decreasing = TRUE),]
```

```
##            Estimate  Std. Error      t value     Pr(>|t|)
## COL    0.6875257478 0.064816319 10.607294012 2.227601e-21
```

```
## MEX    0.6300950078 0.062124067 10.142526733 5.641868e-20
## TUR    0.1169979462 0.022591552  5.178836098 5.160960e-07
## HUN    0.0657887546 0.014758862  4.457576253 1.339881e-05
## OECDE  0.0608869968 0.016460237  3.699035336 2.754444e-04
## POL    0.0571416170 0.009891745  5.776697186 2.676553e-08
## OECD   0.0565869174 0.015141500  3.737206876 2.390295e-04
## SVK    0.0538127180 0.021424435  2.511744979 1.275560e-02
## IRL    0.0537131517 0.022005200  2.440929982 1.546608e-02
## ISL    0.0522388093 0.020435928  2.556224011 1.127783e-02
## LVA    0.0495076507 0.016048052  3.084963320 2.334657e-03
## EST    0.0443833705 0.020238322  2.193036078 2.949381e-02
## ISR    0.0432675012 0.016025826  2.699860874 7.494143e-03
## CZE    0.0422226505 0.018309874  2.306004372 2.207236e-02
## CHL    0.0334916296 0.020320066  1.648204737 1.010265e-01
## KOR    0.0248701186 0.015095836  1.647482067 1.009334e-01
## SVN    0.0212908415 0.019171833  1.110527190 2.683402e-01
## ITA    0.0201469436 0.013357913  1.508240346 1.329748e-01
## USA    0.0154336916 0.021043968  0.733402153 4.641196e-01
## PRT    0.0125589509 0.022935884  0.547567770 5.845620e-01
## G-7    0.0122058098 0.020261793  0.602405206 5.475452e-01
## JPN    0.0111754282 0.021467188  0.520581835 6.031988e-01
## GBR    0.0096913181 0.024436671  0.396589135 6.920675e-01
## FIN    0.0078325799 0.018203943  0.430268322 6.674355e-01
## GRC    0.0067366218 0.027047178  0.249069301 8.035472e-01
## FRA    0.0049815153 0.023424865  0.212659295 8.317961e-01
## NLD    0.0035106463 0.030813165  0.113933324 9.093980e-01
## ESP    0.0029942022 0.030355860  0.098636712 9.215195e-01
## CAN    0.0021419825 0.023529301  0.091034685 9.275506e-01
## DNK    0.0013678495 0.042881660  0.031898241 9.745831e-01
## EU28   0.0009921724 0.049655449  0.019981139 9.840771e-01
## EA19   0.0001024458 0.053819263  0.001903515 9.984830e-01
## AUT   -0.0003468553 0.025314487 -0.013701850 9.890807e-01
## NOR   -0.0006894707 0.020505701 -0.033623367 9.732090e-01
## SWE   -0.0024159928 0.020702851 -0.116698558 9.072089e-01
## LUX   -0.0028502747 0.052235107 -0.054566266 9.565352e-01
## DEU   -0.0039919033 0.035099491 -0.113731087 9.095581e-01
## BEL   -0.0109081216 0.019370561 -0.563128846 5.739394e-01
```

Ideas for extention

My extension will mainly focus on the update of data period. The paper use data from 1983-2007 in table 1 and 2, while I add 2008-2016 period into the results, to estimate the inflation expectation in the post-recession period. Also in the cross-country comparison, the authors use data from 1996-2013, while I can use data from 1996-2016 to make the estimate more power. In the moving-window estimates, I will also add more recent data into the graph.I expect to see a more anchored result of recent data than the original paper because the inflation is more targeted by the central bank of most countries recently. And in the paper I will discuss more about it.