Exercise 8G

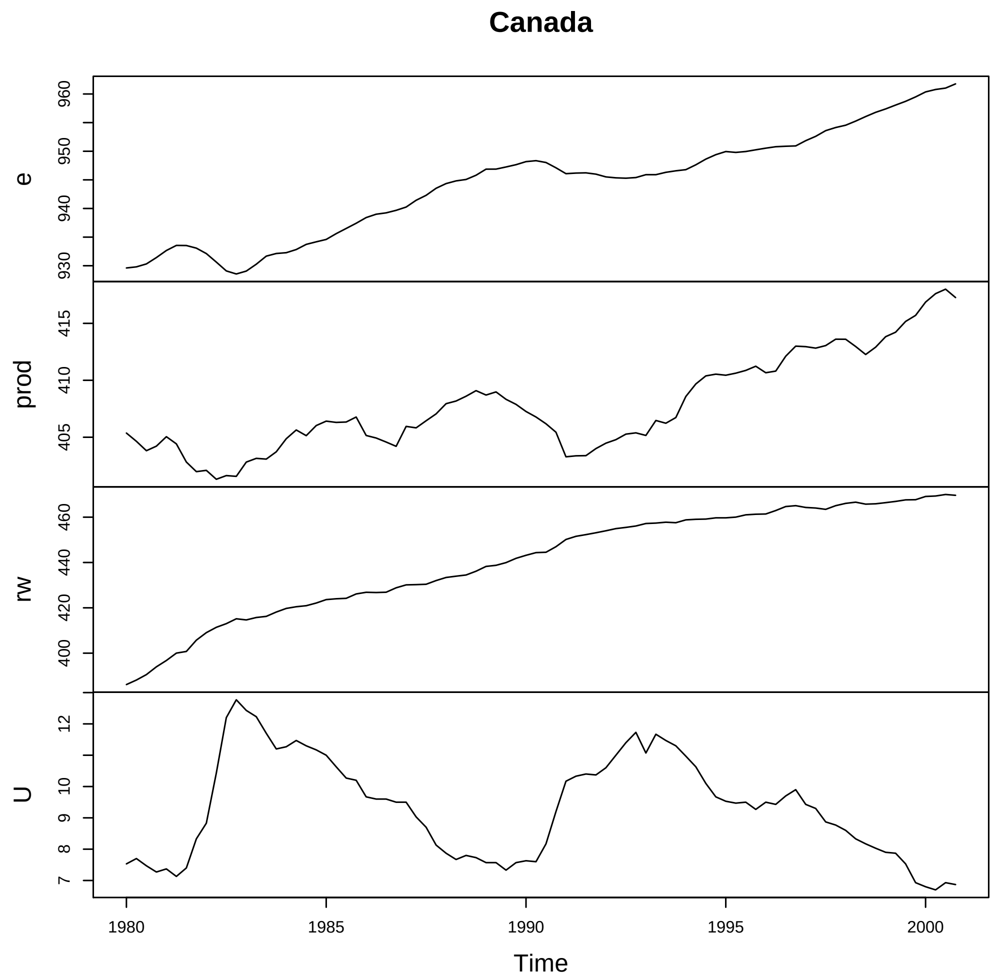
Task Description

1. Set up a VAR (with data of your own choice).
2. Choose the order of the VAR (VAR(xy, ic ="AIC", ...))
3. Test the residuals wrt white noise.
4. Comment on the roots of the characteristic polynomial.
5. Generate forecasts for at least 4 periods into the future.
6. Inspect the MA(infinity) representations of the model.
7. Comment on the impulse response function.
8. Comment on the forecast error decomposition.

Data

- Data description: Canadian labor market data
- Data scource: OECD
- Period: 1stQ 1980 until 4thQ 2000
- The variable **e** stands for employment
- The variable **prod** assigns labour productivity
- The variable **rw** stands for the real wage
- The variable **U** is the unemployment rate

Data Plot



In the plot we can see the similar trend in variables e, prod and rw, which can possibly indicate, that there is a certain ammount of connection between the variables. We will consider this situation more closely later.

VAR model selection

At first, we would like to fit this data with the VAR model. For it, we need to determine the optimal lag-order.To find it we use the function **VARselect**, which gives us several information criteria.

```
VARselect(Canada, lag.max=6)

## $selection
## AIC(n)  HQ(n)  SC(n) FPE(n)
##      3      2      2      3
##
## $criteria
##                   1            2            3            4            5
## AIC(n) -5.832203860 -6.341153584 -6.440077215 -6.228986742 -6.021458099
## HQ(n)  -5.590297749 -5.905722585 -5.811121327 -5.406505966 -5.005452434
## SC(n)  -5.227919545 -5.253441818 -4.868937997 -4.174420073 -3.483463978
## FPE(n)  0.002933674  0.001769561  0.001616452  0.002028286  0.002561839
##                  6
## AIC(n) -5.89342928
## HQ(n)  -4.68389873
## SC(n)  -2.87200771
## FPE(n)  0.00302821
```

To choose the optimal lag, we take the one with the smallest information criteria. As you can see, AIC shows that the third lag is the optimal one. However, we choose a lag 2, according to Schwarz Critertion and Hannan-Quinn Criterion, since they are the most conservative criteria.

Model VAR(2)

So now we can calculate VAR(2) model.

```
mod <- VAR(Canada, p = 2, type = "const")
summary(mod)

##
## VAR Estimation Results:
## =========================
## Endogenous variables: e, prod, rw, U
## Deterministic variables: const
## Sample size: 82
## Log Likelihood: -175.819
## Roots of the characteristic polynomial:
```

```
## 0.995 0.9081 0.9081 0.7381 0.7381 0.1856 0.1429 0.1429
## Call:
## VAR(y = Canada, p = 2, type = "const")
##
##
## Estimation results for equation e:
## ==================================
## e = e.l1 + prod.l1 + rw.l1 + U.l1 + e.l2 + prod.l2 + rw.l2 + U.l2 + const
##
##            Estimate Std. Error t value Pr(>|t|)
## e.l1     1.638e+00  1.500e-01  10.918  < 2e-16 ***
## prod.l1  1.673e-01  6.114e-02   2.736  0.00780 **
## rw.l1   -6.312e-02  5.524e-02  -1.143  0.25692
## U.l1     2.656e-01  2.028e-01   1.310  0.19444
## e.l2    -4.971e-01  1.595e-01  -3.116  0.00262 **
## prod.l2 -1.017e-01  6.607e-02  -1.539  0.12824
## rw.l2    3.844e-03  5.552e-02   0.069  0.94499
## U.l2     1.327e-01  2.073e-01   0.640  0.52418
## const   -1.370e+02  5.585e+01  -2.453  0.01655 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##
## Residual standard error: 0.3628 on 73 degrees of freedom
## Multiple R-Squared: 0.9985,Adjusted R-squared: 0.9984
## F-statistic:  6189 on 8 and 73 DF,  p-value: < 2.2e-16
##
##
## Estimation results for equation prod:
## =====================================
## prod = e.l1 + prod.l1 + rw.l1 + U.l1 + e.l2 + prod.l2 + rw.l2 + U.l2 + const
##
##            Estimate Std. Error t value Pr(>|t|)
## e.l1      -0.17277    0.26977  -0.640  0.52390
## prod.l1    1.15043    0.10995  10.464 3.57e-16 ***
## rw.l1      0.05130    0.09934   0.516  0.60710
## U.l1      -0.47850    0.36470  -1.312  0.19362
```

```
## e.l2       0.38526    0.28688   1.343  0.18346
## prod.l2   -0.17241    0.11881  -1.451  0.15104
## rw.l2     -0.11885    0.09985  -1.190  0.23778
## U.l2       1.01592    0.37285   2.725  0.00805 **
## const   -166.77552  100.43388  -1.661  0.10109
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##
## Residual standard error: 0.6525 on 73 degrees of freedom
## Multiple R-Squared: 0.9787,Adjusted R-squared: 0.9764
## F-statistic: 419.3 on 8 and 73 DF,  p-value: < 2.2e-16
##
##
## Estimation results for equation rw:
## ===================================
## rw = e.l1 + prod.l1 + rw.l1 + U.l1 + e.l2 + prod.l2 + rw.l2 + U.l2 + const
##
##           Estimate Std. Error t value Pr(>|t|)
## e.l1     -0.268833   0.322619  -0.833    0.407
## prod.l1  -0.081065   0.131487  -0.617    0.539
## rw.l1     0.895478   0.118800   7.538 1.04e-10 ***
## U.l1      0.012130   0.436149   0.028    0.978
## e.l2      0.367849   0.343087   1.072    0.287
## prod.l2  -0.005181   0.142093  -0.036    0.971
## rw.l2     0.052677   0.119410   0.441    0.660
## U.l2     -0.127708   0.445892  -0.286    0.775
## const   -33.188339 120.110525  -0.276    0.783
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##
## Residual standard error: 0.7803 on 73 degrees of freedom
## Multiple R-Squared: 0.9989,Adjusted R-squared: 0.9987
## F-statistic:  8009 on 8 and 73 DF,  p-value: < 2.2e-16
##
##
```

```
## Estimation results for equation U:
## ===================================
## U = e.l1 + prod.l1 + rw.l1 + U.l1 + e.l2 + prod.l2 + rw.l2 + U.l2 + const
##
##          Estimate Std. Error t value Pr(>|t|)
## e.l1     -0.58076    0.11563  -5.023 3.49e-06 ***
## prod.l1  -0.07812    0.04713  -1.658 0.101682
## rw.l1     0.01866    0.04258   0.438 0.662463
## U.l1      0.61893    0.15632   3.959 0.000173 ***
## e.l2      0.40982    0.12296   3.333 0.001352 **
## prod.l2   0.05212    0.05093   1.023 0.309513
## rw.l2     0.04180    0.04280   0.977 0.331928
## U.l2     -0.07117    0.15981  -0.445 0.657395
## const   149.78056   43.04810   3.479 0.000851 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##
## Residual standard error: 0.2797 on 73 degrees of freedom
## Multiple R-Squared: 0.9726,Adjusted R-squared: 0.9696
## F-statistic:   324 on 8 and 73 DF,  p-value: < 2.2e-16
##
##
##
## Covariance matrix of residuals:
##              e      prod       rw        U
## e     0.131635 -0.007469 -0.04210 -0.06909
## prod -0.007469  0.425711  0.06461  0.01392
## rw   -0.042099  0.064613  0.60886  0.03422
## U    -0.069087  0.013923  0.03422  0.07821
##
## Correlation matrix of residuals:
##             e     prod      rw       U
## e     1.00000 -0.03155 -0.1487 -0.6809
## prod -0.03155  1.00000  0.1269  0.0763
## rw   -0.14870  0.12691  1.0000  0.1568
## U    -0.68090  0.07630  0.1568  1.0000
```

Summary gives us an overview on which variables and their lags are significant in each equation. Moreover, covariance and correlation matrices of the resduals for four equations are given.

Model stability

To check the stability of VAR model we need to analize the roots of the characteristic equationuse. They must be outside the unit circle. The function **roots()** computes the eigenvalues of the companion matrix and returns by default their moduli. As was proved in our excercises, the equivalent statement to the original check is that the eigenvalues should be inside the unit circle.

```
roots(mod)

## [1] 0.9950338 0.9081062 0.9081062 0.7380565 0.7380565 0.1856381 0.1428889
## [8] 0.1428889
```

As we can see, the model is stable with a constant as deterministic regressor.

White noise test

Now we want to test the residuals wrt white noise.The Ljung-Box Tes is used for this. The Null Hypothesis is White noise.

```
source("mq_R.txt")
mq(resid(mod),10)

## [1] "m,       Q(m) and  p-value:"
## [1] 1.0000000 7.2671553 0.9677615
## [1]  2.000000 29.555391  0.590849
## [1]  3.0000000 42.7113365  0.6887196
## [1]  4.000000 57.972554  0.688294
## [1]  5.0000000 67.5251595  0.8387339
## [1]  6.0000000 77.3307016  0.9188228
## [1]  7.0000000 85.6899697  0.9694438
## [1]   8.0000000 103.2291731   0.9472444
```

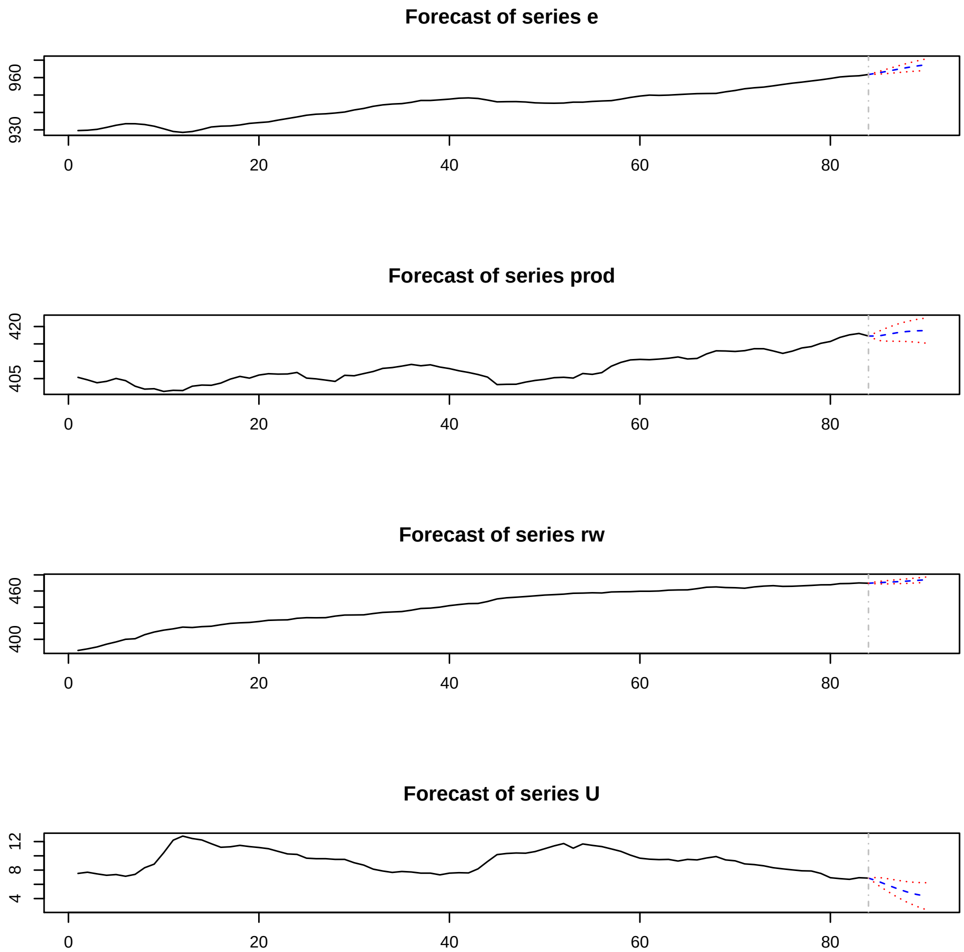
```
## [1]   9.0000000 107.9603599   0.9890493
## [1]  10.0000000 124.9602646   0.9814822
```

Based on the output we don't reject the H0 (residuals are white noise) at 95% significance level.

Forecasting

To do a forecast we use the **predict()** function for the lag 6.

```
plot(predict(mod, n.ahead= 6,interval='confidence'))
```



As you can see from the graphs, the prediction seems to be correct, since it goes in the same direction as a trend. Furthermore, if we look at the confidence intervals for each forecast, we can notice, that for variables prod and U the intervals are larger, which perfectly correlates with their higher volatility in the historical period.

MA(infinity) representation

As we have understood, that our system is stabile, we can look now at the MA(infinity) representation of the resulting VAR model.

```
## , , 1
##
##      [,1] [,2] [,3] [,4]
## [1,]    1    0    0    0
## [2,]    0    1    0    0
## [3,]    0    0    1    0
## [4,]    0    0    0    1
##
## , , 2
##
##            [,1]        [,2]        [,3]        [,4]
## [1,]  1.6378206  0.16727167 -0.06311863  0.26558478
## [2,] -0.1727658  1.15042820  0.05130390 -0.47850131
## [3,] -0.2688329 -0.08106500  0.89547833  0.01213003
## [4,] -0.5807638 -0.07811707  0.01866214  0.61893150
##
## , , 3
##
##            [,1]       [,2]        [,3]       [,4]
## [1,]  2.0191501  0.3491150 -0.14251580  0.6512430
## [2,]  0.1676489  1.1553945 -0.01191318  0.1240154
## [3,] -0.3062245 -0.2169480  0.86759379 -0.1419466
## [4,] -0.8923428 -0.1847587  0.10271259  0.1952708
##
## , , 4
##
##            [,1]       [,2]       [,3]       [,4]
## [1,]  2.2426353  0.5189024 -0.2308082  1.1469646
## [2,]  0.3580192  1.1425299 -0.1143523  0.7415988
## [3,] -0.1780731 -0.3227527  0.8387395 -0.2880998
## [4,] -1.0514599 -0.2807327  0.1763723 -0.2293339
```

In the output we have 4 matrices for lags 0,1,2 and 3. In MA representation columns represent the variables, that cause the shock, and rows are those varibles, that are influenced by it.

In the second matrix we already can see the changes in the variables at lag 1. Here, for example, we can see, that the increase in employment already causes an effect, the negative one, on all other variables. In the third matrix the effect strengthens, as the change in employment itself is higher, and the same situation happens at the lag 4, represented by the last matrix.

Impulse response function

Impulse response functions represent the mechanisms through which shock spread over time. It is useful to look at the impulse response effect explicitly on the graph. Below, you can see the plots of the functions themselves and the corresponding non-cumulated shocks of the variables in response to changes in other variables in different lags. Moreover, there are also calculated lower and higher 95% confidence boundaries of those shocks, obtained by the Bootstrap method.

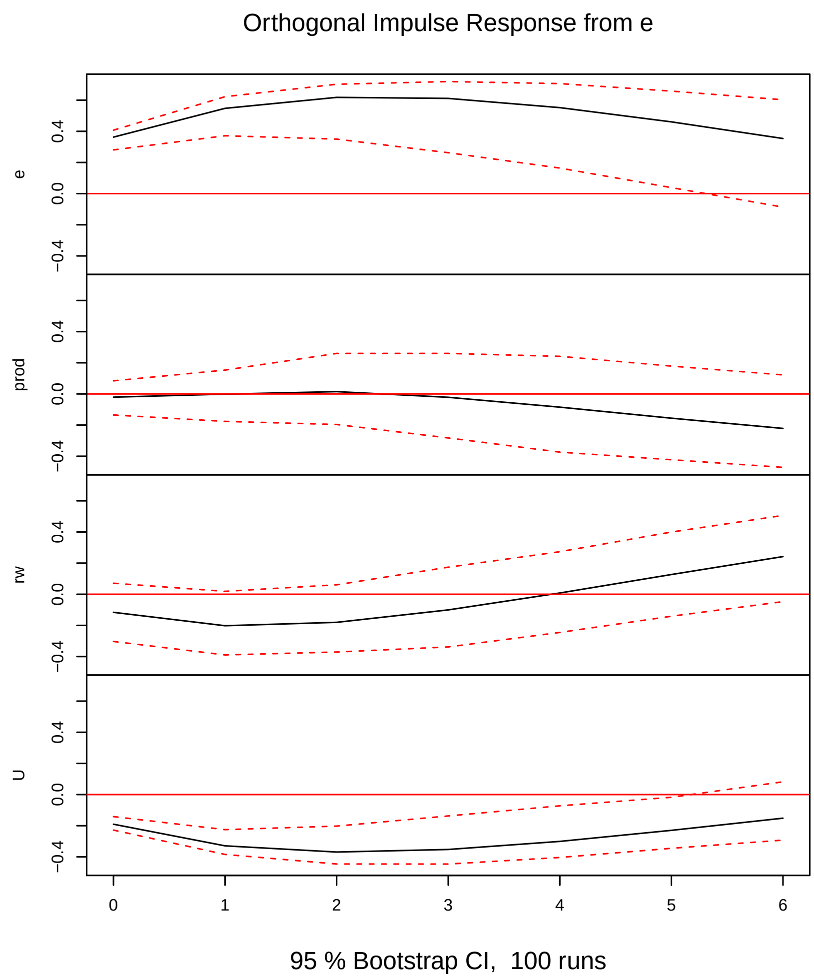
```
##
## Impulse response coefficients
## $e
##               e         prod           rw            U
##  [1,] 0.3628150 -0.020585541 -0.116033519 -0.190420048
##  [2,] 0.5475337 -0.001200947 -0.202083140 -0.329124153
##  [3,] 0.6179181  0.014808436 -0.180277335 -0.369053587
##  [4,] 0.6113563 -0.021571434 -0.100425475 -0.352501745
##  [5,] 0.5520475 -0.084914238  0.008049928 -0.300681928
##  [6,] 0.4606940 -0.155700530  0.126762159 -0.229617289
##  [7,] 0.3538296 -0.221442354  0.241833321 -0.151593876
##  [8,] 0.2437632 -0.274945401  0.343821736 -0.075179522
##  [9,] 0.1390056 -0.313059778  0.427131741 -0.005842792
##
## $prod
##               e      prod          rw           U
##  [1,] 0.0000000 0.6521403  0.09541606  0.01533867
##  [2,] 0.1071358 0.7477963  0.03276331 -0.03966904
##  [3,] 0.2240629 0.7542449 -0.06087546 -0.10769297
##  [4,] 0.3339673 0.7455539 -0.13486988 -0.16976603
##  [5,] 0.4325320 0.7284313 -0.18399188 -0.22353359
##  [6,] 0.5176453 0.7044854 -0.20923845 -0.26772154
##  [7,] 0.5883155 0.6742212 -0.21269765 -0.30182766
##  [8,] 0.6443195 0.6383660 -0.19700793 -0.32595761
```

```
##  [9,] 0.6860526 0.5979989 -0.16507378 -0.34065646
##
## $rw
##                e         prod        rw          U
##  [1,]  0.00000000  0.000000000 0.7656960 0.01392474
##  [2,] -0.04463148  0.032620180 0.6858331 0.02290799
##  [3,] -0.10005539 -0.007394995 0.6623365 0.08136571
##  [4,] -0.16075773 -0.077232540 0.6382078 0.13185418
##  [5,] -0.21764015 -0.124757647 0.6103100 0.17342028
##  [6,] -0.26207007 -0.148110654 0.5752906 0.20225555
##  [7,] -0.29035880 -0.151072129 0.5306628 0.21716396
##  [8,] -0.30225678 -0.138649366 0.4769756 0.21927750
##  [9,] -0.29954466 -0.116006104 0.4164717 0.21084913
##
## $U
##               e        prod           rw           U
##  [1,] 0.00000000  0.00000000  0.000000000  0.20376705
##  [2,] 0.05411743 -0.09750280  0.002471701  0.12611784
##  [3,] 0.13270186  0.02527026 -0.028924044  0.03978976
##  [4,] 0.23371359  0.15111339 -0.058705253 -0.04673068
##  [5,] 0.33598154  0.24344474 -0.086137815 -0.12589580
##  [6,] 0.42502583  0.30133081 -0.100655507 -0.18914406
##  [7,] 0.49382949  0.32837608 -0.097247777 -0.23358628
##  [8,] 0.54042395  0.33083853 -0.075656899 -0.25969750
##  [9,] 0.56601402  0.31551307 -0.038340828 -0.26979650
##
##
## Lower Band, CI= 0.95
## $e
##                e       prod           rw          U
##  [1,]  0.28639004 -0.1475179 -0.287235123 -0.2186989
##  [2,]  0.37378670 -0.2068527 -0.375362340 -0.3701172
##  [3,]  0.36795501 -0.2531018 -0.382030348 -0.4210042
##  [4,]  0.30312800 -0.3164414 -0.343102686 -0.4257663
##  [5,]  0.19830150 -0.4064974 -0.237283107 -0.3831880
##  [6,]  0.07008910 -0.4972562 -0.160028800 -0.3551088
##  [7,] -0.06458291 -0.5610555 -0.088458545 -0.3054288
```

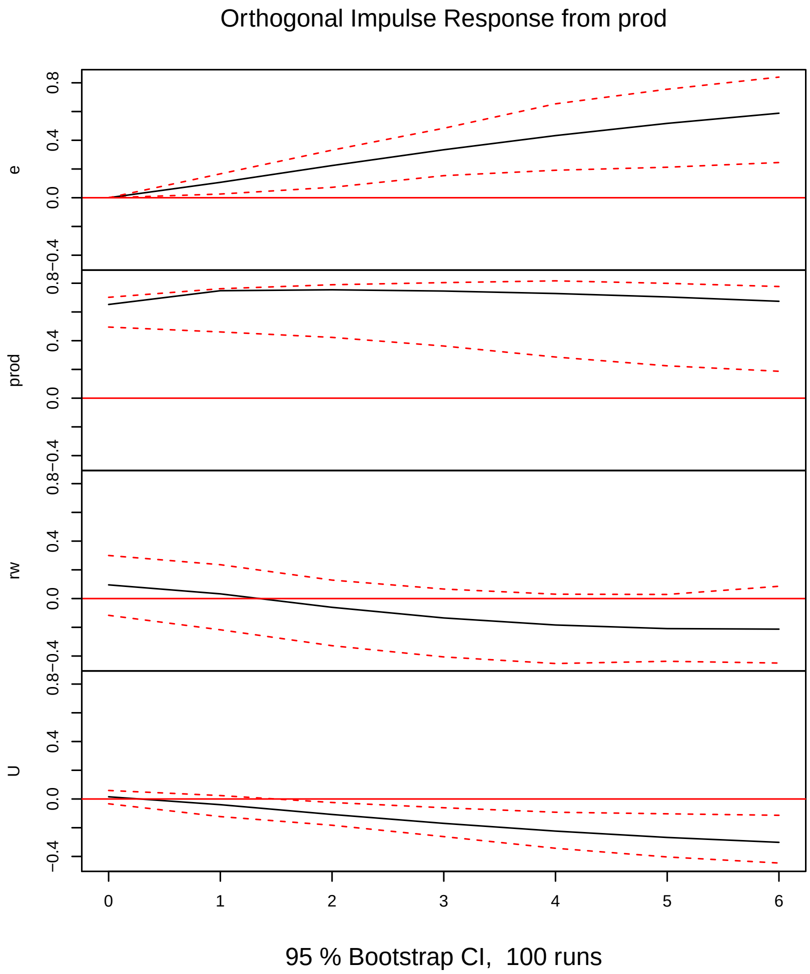
```
##  [8,] -0.18711866 -0.5811331  0.007857494 -0.2447020
##  [9,] -0.28479759 -0.5593190  0.104831226 -0.1850344
##
## $prod
##                e       prod          rw           U
##  [1,] 0.00000000 0.49369606 -0.08243307 -0.03747420
##  [2,] 0.03115768 0.47595255 -0.26809787 -0.09684103
##  [3,] 0.08287128 0.44207821 -0.38672829 -0.20819158
##  [4,] 0.14795430 0.39977274 -0.45044201 -0.29574537
##  [5,] 0.22586401 0.33061670 -0.48117948 -0.35098574
##  [6,] 0.27660043 0.26749837 -0.48319534 -0.41177487
##  [7,] 0.30360228 0.19755876 -0.47016153 -0.45452959
##  [8,] 0.28172033 0.13368662 -0.47549491 -0.46944044
##  [9,] 0.25480801 0.08171218 -0.41222766 -0.48105936
##
## $rw
##                e        prod        rw           U
##  [1,]  0.0000000  0.00000000 0.6091545 -0.02516556
##  [2,] -0.1221585 -0.08720608 0.4180590 -0.04102597
##  [3,] -0.2031698 -0.12844347 0.4036064  0.01418807
##  [4,] -0.2970556 -0.24831411 0.3068834  0.03952778
##  [5,] -0.3795984 -0.29606886 0.2799253  0.05279085
##  [6,] -0.4382231 -0.29391592 0.2528983  0.05955454
##  [7,] -0.4820063 -0.28719169 0.2251312  0.05075139
##  [8,] -0.4861886 -0.26238560 0.1861498  0.03776877
##  [9,] -0.4459515 -0.22305634 0.1304796  0.02334963
##
## $U
##                 e         prod         rw           U
##  [1,]  0.00000000  0.000000000  0.0000000  0.15908833
##  [2,] -0.01666340 -0.214742420 -0.1127869  0.04174269
##  [3,]  0.01558900 -0.155434401 -0.1541457 -0.05929808
##  [4,]  0.07382926 -0.066164216 -0.2344921 -0.15928557
##  [5,]  0.11496922 -0.018800929 -0.2424581 -0.24240791
##  [6,]  0.13228136 -0.009730567 -0.2617095 -0.29652277
##  [7,]  0.14999394 -0.022700132 -0.2390848 -0.34190136
##  [8,]  0.15714083 -0.035842126 -0.1914339 -0.37451573
```

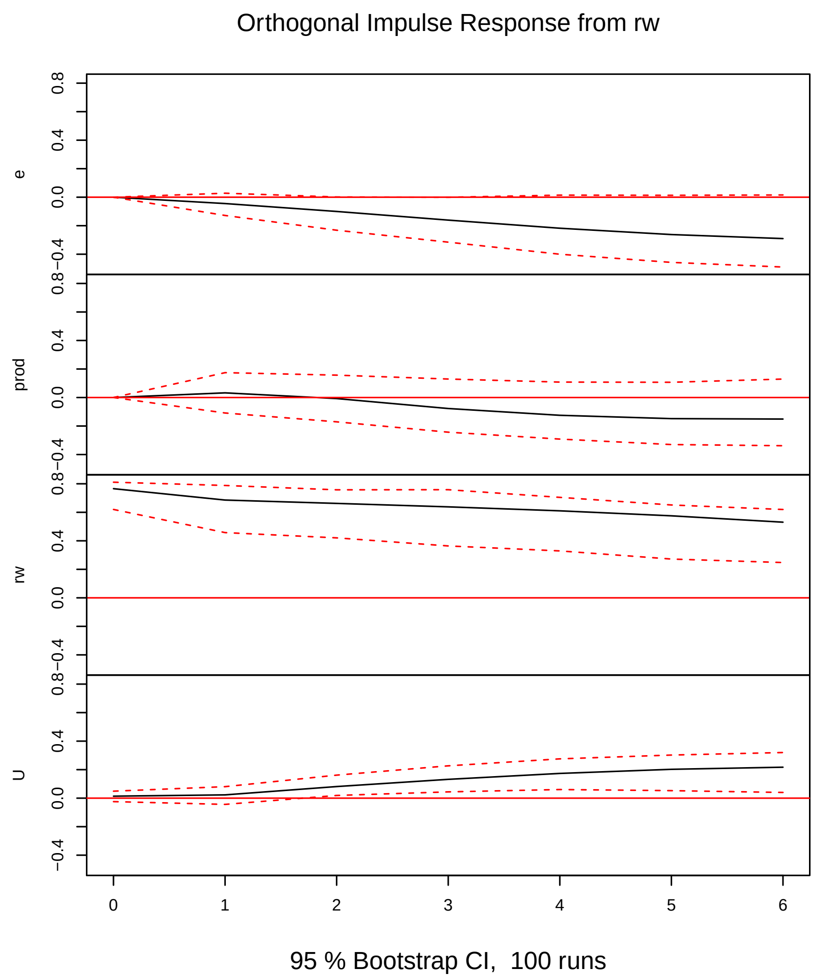
```
##  [9,]  0.15633103 -0.052463526 -0.1543876 -0.37883156
##
##
## Upper Band, CI= 0.95
## $e
##                e       prod         rw            U
##  [1,] 0.3858983 0.09251166 0.05394028 -0.119811957
##  [2,] 0.5769534 0.20537078 0.05056961 -0.207981850
##  [3,] 0.6835429 0.29343562 0.12041550 -0.201314433
##  [4,] 0.7081563 0.28615517 0.27047942 -0.165921425
##  [5,] 0.7091980 0.23166066 0.36212508 -0.096563340
##  [6,] 0.6679868 0.19276917 0.43304654  0.003222328
##  [7,] 0.6173997 0.14040435 0.53016954  0.093750012
##  [8,] 0.5526918 0.11351689 0.62917575  0.152196178
##  [9,] 0.4788250 0.09417418 0.70470394  0.208262170
##
## $prod
##                e      prod         rw           U
##  [1,] 0.0000000 0.7029605 0.29632493  0.05928387
##  [2,] 0.1799719 0.8760278 0.28571980  0.01571962
##  [3,] 0.3708341 0.9286927 0.17960475 -0.02511178
##  [4,] 0.5289311 0.9151561 0.13397404 -0.06735227
##  [5,] 0.6632719 0.9003106 0.10065088 -0.09760610
##  [6,] 0.7651542 0.8720370 0.09216785 -0.11773307
##  [7,] 0.8288038 0.8224192 0.12685555 -0.13536629
##  [8,] 0.8871343 0.7690650 0.19639820 -0.12986632
##  [9,] 0.9278341 0.7180775 0.24544164 -0.11203409
##
## $rw
##                   e       prod        rw          U
##  [1,]  0.000000000 0.00000000 0.8129993 0.05721088
##  [2,]  0.030887574 0.13372951 0.7816209 0.09354651
##  [3,]  0.022733579 0.13171708 0.7459925 0.15307825
##  [4,]  0.013860873 0.08901376 0.6915052 0.20859490
##  [5,] -0.004247637 0.06895859 0.6664188 0.26453841
##  [6,] -0.003324552 0.07631098 0.6450947 0.29727226
##  [7,]  0.003025946 0.11560001 0.6111682 0.32528610
```

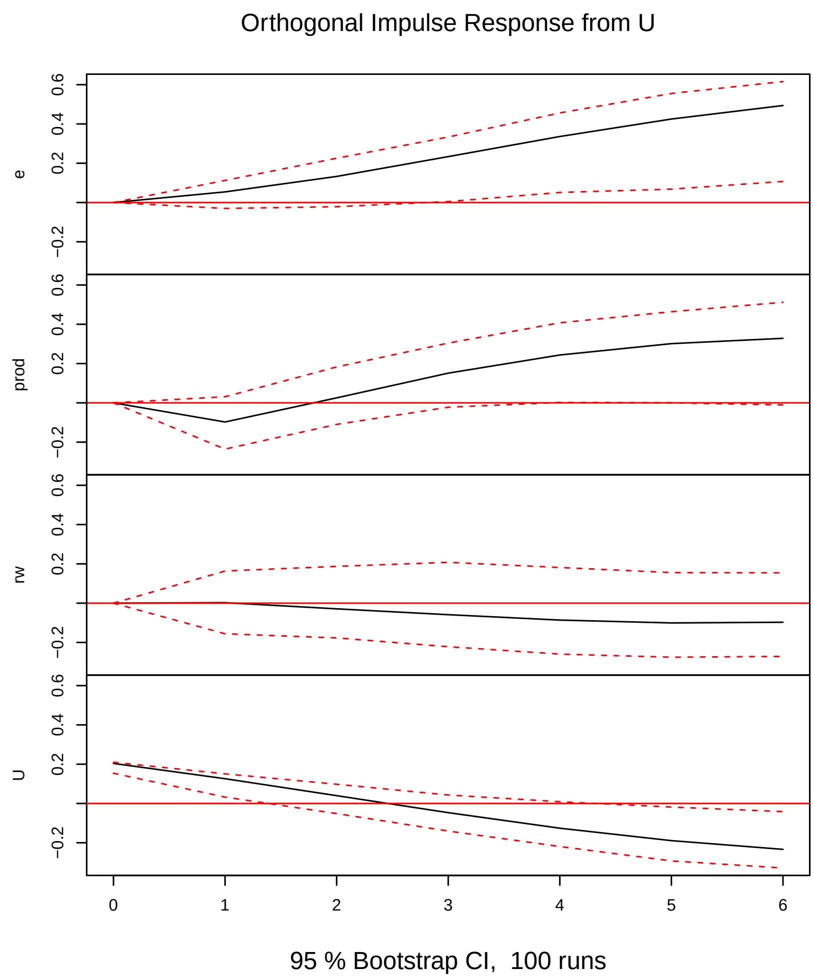
```
## [8,]  0.025550897 0.16070324 0.5649239 0.31329934
## [9,]  0.044461326 0.20413142 0.5099380 0.28397757
##
## $U
##               e        prod        rw           U
## [1,] 0.0000000 0.000000000 0.0000000  0.21847555
## [2,] 0.1163201 0.006797988 0.1631957  0.16107934
## [3,] 0.2333405 0.198668919 0.2016889  0.10092067
## [4,] 0.3556895 0.347022962 0.1989223  0.02289429
## [5,] 0.5210439 0.433768631 0.1917386 -0.02656959
## [6,] 0.6235273 0.479680947 0.1936197 -0.05364726
## [7,] 0.7098216 0.511895242 0.2082193 -0.06411479
## [8,] 0.7781471 0.503135793 0.2093661 -0.05964698
## [9,] 0.8228030 0.470733505 0.2438541 -0.05242229
```



Orthogonal Impulse Response from e
e
prod
rw
U
0.4
0.0
−0.4
0
1
2
3
4
5
6
95 % Bootstrap CI, 100 runs


Orthogonal Impulse Response from prod
e
prod
rw
U
0.8
0.4
0.0
−0.4
0 1 2 3 4 5 6
95 % Bootstrap CI, 100 runs


Orthogonal Impulse Response from rw
e
prod
rw
U
0.8
0.4
0.0
−0.4
0
1
2
3
4
5
6
95 % Bootstrap CI, 100 runs



The increase in employment at the time period 0 can cause the further slight increase in it for several periods, but the level can stay almost the same. From confidence intervals we can also see, that till the 6th lag the effect can completely disappear, which makes perfect economic sense.

Furthermore, if we look at the graph of changes in unemployment, we can see, that it is mirroring the employment, which is quite logical, as, for instance, the increasing employment in 2nd lag is accompanied by the enlarging decrease in unemployment.

Concerning the measure of productivity, we can also see, that at first the change in employment hardly influences it, but with time the productivity decreases, as the market is overflowing with the labour and it may cause worsening of work quality. Moreover, we can notice, that according to the confidence intervals, the actual effect of employment on productivity can be overall positive or negative, but the trend will be the same.

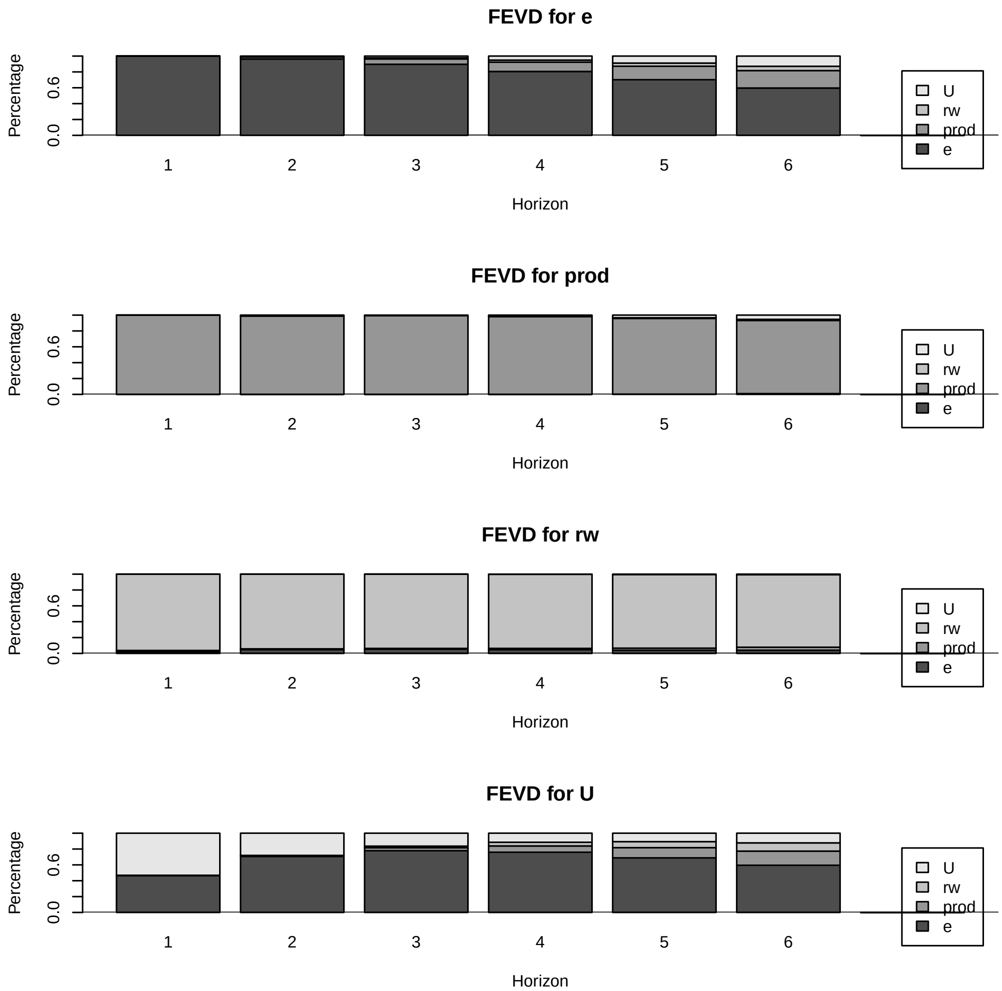
In regards to the real wage it is seen, that the increase in employment can at first lessen the level of real wage, but in time it go higher, as the wages are adjusted to the overall changes in economy.

Forecast error decomposition

Forecast error decomposition is based on the orthogonalized MA representation of VAR, and shows the contribution of one variable to the lagged forecast error variance in some other variable. Below we can see the contributions of the change in different variables to the one with respect to the time period and their graphical representation.

```
## $e
##              e       prod          rw           U
## [1,] 1.0000000 0.00000000 0.000000000 0.000000000
## [2,] 0.9633815 0.02563062 0.004448081 0.006539797
## [3,] 0.8961692 0.06797131 0.013226872 0.022632567
## [4,] 0.8057174 0.11757589 0.025689192 0.051017495
## [5,] 0.7019003 0.16952744 0.040094324 0.088477959
## [6,] 0.5968810 0.22006695 0.053906276 0.129145745
## [7,] 0.5005562 0.26663487 0.065183552 0.167625374
## [8,] 0.4185475 0.30793933 0.073035978 0.200477222
##
## $prod
##                 e      prod          rw           U
## [1,] 0.0009954282 0.9990046 0.000000000 0.000000000
## [2,] 0.0004271364 0.9889540 0.001068905 0.009549939
## [3,] 0.0004117451 0.9923920 0.000714736 0.006481511
## [4,] 0.0005161018 0.9808528 0.003294103 0.015337005
## [5,] 0.0030112576 0.9554064 0.008196778 0.033385578
## [6,] 0.0095876800 0.9233896 0.013127373 0.053895335
```

```
## [7,] 0.0202450680 0.8908636 0.016724005 0.072167323
## [8,] 0.0338608030 0.8612441 0.018660981 0.086234112
##
## $rw
##                e        prod        rw            U
## [1,] 0.02211315 0.014952942 0.9629339 0.000000e+00
## [2,] 0.04843396 0.009077935 0.9424827 5.449178e-06
## [3,] 0.05435699 0.008694159 0.9364211 5.277244e-04
## [4,] 0.04758894 0.015753861 0.9345505 2.106688e-03
## [5,] 0.03957649 0.026911791 0.9287321 4.779622e-03
## [6,] 0.03964695 0.038484795 0.9142068 7.661476e-03
## [7,] 0.05284433 0.047743039 0.8897691 9.643486e-03
## [8,] 0.07968665 0.053293594 0.8568370 1.018273e-02
##
## $U
##               e        prod          rw         U
## [1,] 0.4636211 0.003008244 0.002479203 0.5308915
## [2,] 0.7068777 0.008843922 0.003513667 0.2807647
## [3,] 0.7787875 0.037185141 0.020355796 0.1636716
## [4,] 0.7596609 0.079197860 0.046371393 0.1147699
## [5,] 0.6886156 0.128139114 0.076164201 0.1070811
## [6,] 0.5954733 0.178008698 0.103964487 0.1225535
## [7,] 0.5026125 0.224371197 0.125721703 0.1472946
## [8,] 0.4229416 0.264861489 0.140012873 0.1721840
```



On the first chart we can see, that apart from employment itself, in time some uncertainty is brought by the labour productivity. On the other hand, for productivity itself almost no other variables cause the uncertainty. The same situation is observed in real wage. However, for unemployment employment plays a huge role, and in time the effect of unemployment itself is becoming less and less, while productivity starts showing some influence on unemployment forecast uncertainty.