

ECON7350: Applied Econometrics for Macroeconomics and Finance

Tutorial 12: Multivariate Processes - III

At the end of this tutorial you should be able to:

- Use R to construct an adequate set of VECM models;
- Use R to obtain inference on equilibrium relationships from VECM models;
- Use R to obtain inference on dynamic relationships from identified structural VECMs.

Assignment Project Exam Help Problems with Solutions

The data file `termstructure.csv` contains data on four Australian interest rates: the 5 year (`i5y`) and 3 year (`i3y`) Treasury Bond (capital market) rates, along with the 180 day (`i180d`) and 90 day (`i90d`) Bank Accepted Bill (money market) rates. The data consists of annualized monthly rates for the period June 1992 to August 2010 ($T = 219$).

1. Consider a VAR model of the multivariate process $\{\mathbf{x}_t\}$, where

$$\mathbf{x}_t = (i90d_t, i180d_t, i3y_t, i5y_t)'$$

Construct an adequate set of VAR models specified by the lag length p .

Solution For this tutor we use the following packages.

```
library(dplyr)
library(zoo)
library(vars)
library(urca)
```

Notice that we have loaded a new package called `urca`. This package provides additional functionality for working with VECMs. It is designed to be easily integrated with the `vars` package. Next, we load the data, which is the same one used in Tutorial 6.

```

mydata <- read.delim("term_structure.csv", header = TRUE, sep = ",")

dates <- as.yearmon(mydata$obs, format = "%YM%m")
T <- length(dates)
n <- 4

i90d <- mydata$I90D[-T]
i180d <- mydata$I180D[-T]
i3y <- mydata$I3Y[-T]
i5y <- mydata$I5Y[-T]
x <- cbind(i90d, i180d, i3y, i5y)

```

To construct an adequate set of VAR models, we may follow the same approach as in Tutorial 10 and consider VARs specified by $p = 1, \dots, 20$.

```

VAR_est <- list()
ic_var <- matrix(nrow = 20, ncol = 3)
colnames(ic_var) <- c("p", "aic", "bic")
for (p in 1:20)
{
  VAR_est[[p]] <- VAR(x, p)
  ic_var[p,] <- c(p, AIC(VAR_est[[p]]),
                 BIC(VAR_est[[p]]))
}

```

```

ic_aic_var <- ic_var[order(ic_var[,2]),]
ic_bic_var <- ic_var[order(ic_var[,3]),]

```

AIC and BIC both rank VARs specified by $p = 2, 3, 4, 5$, in the top five, while the rest are quite a bit inferior. We proceed with this set and check the residuals. Note that the `vars` package provides a few different tests through the `serial.test` function; we will use the LM test by setting `type = "BG"`, but other tests are just as valid.

```

adq_set_var <- as.matrix(ic_var[2:5,])
adq_idx_var <- c(2:5)

nmods <- length(adq_idx_var)
for (i in 1:nmods)
{
  p <- adq_idx_var[i]
  print(paste0("Checking VAR(", p, ")"))
  print(serial.test(VAR_est[[p]], lags.bg = 1,
                    type = "BG"))
}

```

```
## [1] "Checking VAR(2)"
```

```
##
## Breusch-Godfrey LM test
##
## data: Residuals of VAR object VAR_est[[p]]
## Chi-squared = 61.971, df = 16, p-value = 2.428e-07
##
## [1] "Checking VAR(3)"
##
## Breusch-Godfrey LM test
##
## data: Residuals of VAR object VAR_est[[p]]
## Chi-squared = 23.762, df = 16, p-value = 0.09484
##
## [1] "Checking VAR(4)"
##
## Breusch-Godfrey LM test
##
## data: Residuals of VAR object VAR_est[[p]]
## Chi-squared = 30.39, df = 16, p-value = 0.01608
##
## [1] "Checking VAR(5)"
##
## Breusch-Godfrey LM test
##
## data: Residuals of VAR object VAR_est[[p]]
## Chi-squared = 15.403, df = 16, p-value = 0.4953
```

Assignment Project Exam Help
<https://tutorcs.com>
 WeChat: cstutorcs

Autocorrelations appear to be quite high for the VAR(2), so we remove it from the set. For $p = 4$, the p-value is lower than for $p = 3$ and $p = 5$. This is odd (likely due to the small-sample approximation error associated with the asymptotic sampling distribution of the test statistic), so we won't remove $p = 4$ from the set, but will proceed with caution.

```
adq_set_var <- as.matrix(ic_var[3:5,])
adq_idx_var <- c(3:5)
```

The final adequate set consists of VARs with $p = 3, 4, 5$.

-
2. Using the adequate set constructed in Question 1, implement Johansen's trace test to obtain inference on possible ranks $r = \text{rank } \mathbf{A}(1)$. What does this suggest about possibly expanding the adequate set of VAR models to include restricted VECMs?
-

Solution We use the function `roots` to ascertain the stability of the estimated VARs.

```
nmods <- length(adq_idx_var)
for (i in 1:nmods)
{
  p <- adq_idx_var[i]
  print(paste0("VAR(", p,
               "): Maximum absolute eigenvalue is ",
               max(vars::roots(VAR_est[[p]]))))
}
```

```
## [1] "VAR(3): Maximum absolute eigenvalue is 0.964167475107657"
## [1] "VAR(4): Maximum absolute eigenvalue is 0.972821252519885"
## [1] "VAR(5): Maximum absolute eigenvalue is 0.968410474796044"
```

We have at least one root that is close to a unit root, so we might consider imposing rank restrictions using the VECM representation. The package `urca` provides functions for working with VECMs, and in particular, the function `ca.jo`, which implements Johansen's *trace* and *maximum eigenvalue* tests. We will focus only on the *trace* test in this exercise.

```
nmods <- length(adq_idx_var)
for (i in 1:nmods)
{
  p <- adq_idx_var[i]
  print(paste0("VAR(", p, ")"))
  print(summary(ca.jo(VAR_est[[p]], type = "trace", K = p)))
}
```

```
## [1] "VAR(3)"
##
## #####
## # Johansen-Procedure #
## #####
##
## Test type: trace statistic , with linear trend
##
## Eigenvalues (lambda):
## [1] 0.22743493 0.13239956 0.04716104 0.02685854
##
## Values of teststatistic and critical values of test:
##
##          test 10pct  5pct  1pct
## r <= 3 |   5.85  6.50  8.18 11.65
## r <= 2 |  16.24 15.66 17.95 23.52
## r <= 1 |  46.78 28.71 31.52 37.22
## r = 0  | 102.25 45.23 48.28 55.43
```

```

##
## Eigenvectors, normalised to first column:
## (These are the cointegration relations)
##
##           i90d.l3  i180d.l3      i3y.l3      i5y.l3
## i90d.l3    1.00000000  1.000000  1.00000000  1.000000
## i180d.l3  -1.02836547 -1.913208 -0.9617405 -1.254801
## i3y.l3     -0.04569791  3.878559  0.8823878 -1.635869
## i5y.l3      0.08217474 -3.096674 -0.7591605  2.506959
##
## Weights W:
## (This is the loading matrix)
##
##           i90d.l3  i180d.l3      i3y.l3      i5y.l3
## i90d.d    -0.247157282  0.1060910 -0.15946039  0.002619658
## i180d.d   -0.010621572  0.1332666 -0.20711509  0.002308654
## i3y.d       0.019675931 -0.1111955 -0.09329906 -0.022978060
## i5y.d       0.002499526 -0.0593836 -0.07149273 -0.028243355
##
## [1] "VAR(4)"
##
## #####
## # Johansen procedure #
## #####
##
## Test type: trace statistic , with linear trend
##
## Eigenvalues (lambda):
## [1] 0.16568914 0.12614410 0.04160551 0.01952826
##
## Values of teststatistic and critical values of test:
##
##           test 10pct  5pct  1pct
## r <= 3 |   4.22   6.50   8.18 11.65
## r <= 2 |  13.31  15.66  17.95 23.52
## r <= 1 |  42.17  28.71  31.52 37.22
## r = 0  |  80.94  45.23  48.28 55.43
##
## Eigenvectors, normalised to first column:
## (These are the cointegration relations)
##
##           i90d.l4  i180d.l4      i3y.l4      i5y.l4
## i90d.l4    1.0000000  1.000000  1.00000000  1.0000000
## i180d.l4  -0.9827447 -1.735731 -1.0481294 -0.8791605
## i3y.l4     -0.2966825  2.879389  1.0305366 -1.2888283

```

Assignment Project Exam Help

<https://tutorcs.com>

WeChat: cstutorcs

```

## i5y.l4      0.2875465 -2.246690 -0.8883264  1.6090033
##
## Weights W:
## (This is the loading matrix)
##
##           i90d.l4  i180d.l4      i3y.l4      i5y.l4
## i90d.d  -0.18268163  0.1745860 -0.17796131 -0.007409865
## i180d.d  0.01427081  0.2359865 -0.21852007 -0.011824157
## i3y.d    0.10518344 -0.1544402 -0.08230806 -0.039701873
## i5y.d    0.05067937 -0.1112037 -0.03756652 -0.047303973
##
## [1] "VAR(5)"
##
## #####
## # Johansen-Procedure #
## #####
##
## Test type: trace statistic , with linear trend
##
## Eigenvalues (lambda):
## [1] 0.21606417 0.12006813 0.03852472 0.01816211
##
## Values of test statistic and critical values of test:
##
##           test 10pct  5pct  1pct
## r <= 3 |  3.90  6.30  8.18 11.65
## r <= 2 | 12.27 15.66 17.95 23.52
## r <= 1 | 39.52 28.71 31.52 37.22
## r = 0  | 91.37 45.23 48.28 55.43
##
## Eigenvectors, normalised to first column:
## (These are the cointegration relations)
##
##           i90d.l5  i180d.l5      i3y.l5      i5y.l5
## i90d.l5  1.0000000  1.000000  1.000000  1.0000000
## i180d.l5 -0.8383998 -1.456892 -1.112462 -0.8787942
## i3y.l5   -0.9008697  1.589088  1.180961 -0.9015127
## i5y.l5    0.7697870 -1.200062 -1.003868  1.1854807
##
## Weights W:
## (This is the loading matrix)
##
##           i90d.l5  i180d.l5      i3y.l5      i5y.l5
## i90d.d  -0.34125140  0.25553012 -0.16779632 -0.01224064
## i180d.d -0.22821103  0.40831876 -0.19276538 -0.01833428

```

Assignment Project Exam Help

<https://tutorcs.com>

WeChat: cs_tutorcs

```
## i3y.d      0.17553392 -0.12060757 -0.04655491 -0.05021051
## i5y.d      0.07376629 -0.09170368  0.01337062 -0.05555563
```

Let us denote by $VECM(p, r)$ a specification corresponding to a $VAR(p)$, but with rank $A(1) = r$ imposed. Recall that in the VECM representation of a $VAR(p)$, there are actually only $p - 1$ lags in the differenced series.

For each VAR with $p = 3, 4, 5$, we obtain nearly identical inference regarding r . Namely, we can sequentially test:

1. (test 1)] $H_0 : r = 0$ against $H_1 : r \geq 1$; reject H_0 at 1%;
2. (test 2)] $H_0 : r = 1$ against $H_1 : r \geq 2$; reject H_0 at 1%;
3. (test 3)] $H_0 : r = 2$ against $H_1 : r \geq 3$; fail to reject H_0 at 5%;

Note that sequentially carrying out the *trace* test as done above can be rigorously justified (left as an exercise). However, the reasoning is not as clear when the *maximum eigenvalue* test is used. More importantly, when we fail to reject $H_0 : r = 2$ against $H_1 : r \geq 3$ in “test 3”, this *not* evidence that $r = 2$ exactly since there is *no justification* to accept H_0 ; the only reasonable conclusion is that a VECM with $r = 2$ is *not* empirically distinguishable from VECMs with $r = 3$ or $r = 4$.

However, since we cannot reject specifications with $r < 4$, this suggests that it may be *reasonable* to consider VECMs with $r = 2$ and $r = 3$ as alternative specifications to each unrestricted $VAR(p)$, which in our notation is equivalent to the $VECM(p, 4)$.

-
3. Construct an adequate set of VECM models specified by the lag length p and rank r .
-

Solution There is not a huge computational cost to consider a general class of VECM models with $p = 3, \dots, 6$ and $r = 0, \dots, 4$, so this is what we’ll do. The information obtained in Questions 1 and 2—starting with a class of unrestricted VARs, then using Johansen’s trace test to obtain inference on r —can be useful to either reduce the overall number of VECMs we search over or to refine the adequate set of VECMs we end up constructing using the general search.

Note: we do not include specifications with $p = 2$ because we have already determined in the unrestricted $VAR(2)$ case that residuals are like to be substantially autocorrelated. Since VAR residuals and VECM residuals are equivalent, the same conclusion must hold for restricted VECMs as well, so it would be redundant to consider a $VECM(2, r)$.

However, it is reasonable to consider slightly higher lag orders (i.e., $p > 5$) because a

restricted $\text{VECM}(p, r)$ with $r < 4$ would be more parsimonious than its unrestricted VAR counterpart. Therefore, it is possible that, say an unrestricted $\text{VAR}(6)$, was eliminated based on the fit vs parsimony tradeoff, but a $\text{VECM}(6, 3)$ could be competitive with other specifications by virtue of being more parsimonious. The AIC/BIC criteria should be able to guide us on this!

The `vars` package provides the function `vec2var` which converts output from the `ca.jo` function to an estimated VAR (with restrictions on the coefficients). This is very handy because it allows us, among other things, to easily compare AIC/BIC values across VAR and restricted VECM specifications.

```
VECM_est <- list()
ic_vecm <- matrix(nrow = 4 * (1 + n), ncol = 4)
colnames(ic_vecm) <- c("p", "r", "aic", "bic")
i <- 0
for (p in 3:6)
{
  for (r in 0:n)
  {
    i <- i + 1
    if (r == n)
    {
      VECM_est[[i]] <- VAR(x, p)
    }
    else if (r == 0)
    {
      VECM_est[[i]] <- VAR(diff(x), p - 1)
    }
    else
    {
      VECM_est[[i]] <- vec2var(ca.jo(x, K = p), r)
    }
    ic_vecm[i,] <- c(p, r, AIC(VECM_est[[i]]),
                    BIC(VECM_est[[i]]))
  }
}

ic_aic_vecm <- ic_vecm[order(ic_vecm[,3]),][1:10,]
ic_bic_vecm <- ic_vecm[order(ic_vecm[,4]),][1:10,]

ic_int_vecm <- intersect(as.data.frame(ic_aic_vecm),
                        as.data.frame(ic_bic_vecm))
```

The intersecting set has combinations of $p = 3, 4$ with $r = 2, 3, 4$. Since AIC and BIC values look comparable we will proceed with this as the adequate set. It would not be unreasonable to also include VARs with $p = 5$ since we previously found an unrestricted $\text{VAR}(5)$ to be comparable to an unrestricted $\text{VAR}(3)$ and $\text{VAR}(4)$, but

we will keep it simple.

Again, since we have already looked at unrestricted VAR(3) and VAR(4) model residuals, there is no particular reason to be concerned about VECM(3, r) and VECM(4, r) residuals, but let us do a final check just for completeness.

```
adq_set_vecm <- as.matrix(arrange(as.data.frame(
                                ic_int_vecm), p, r))
adq_idx_vecm <- match(data.frame(t(adq_set_vecm[, 1:2])),
                     data.frame(t(ic_vecm[, 1:2])))

nmods <- length(adq_idx_vecm)
for (i in 1:nmods)
{
  p <- adq_set_vecm[i, 1]
  r <- adq_set_vecm[i, 2]
  print(paste0("Checking VECM(", p, ", ", r, ")"))
  print(serial.test(VECM_est[[adq_idx_vecm[i]]],
                    lags.bg = 1,
                    type = "BG"))
}
```

```
## [1] "Checking VECM(3, 2)"
##
## Breusch-Godfrey LM test
##
## data: Residuals of VAR object VECM_est[[adq_idx_vecm[i]]]
## Chi-squared = 22.798, df = 16, p-value = 0.1192
##
## [1] "Checking VECM(3, 3)"
##
## Breusch-Godfrey LM test
##
## data: Residuals of VAR object VECM_est[[adq_idx_vecm[i]]]
## Chi-squared = 23.752, df = 16, p-value = 0.09506
##
## [1] "Checking VECM(3, 4)"
##
## Breusch-Godfrey LM test
##
## data: Residuals of VAR object VECM_est[[adq_idx_vecm[i]]]
## Chi-squared = 23.762, df = 16, p-value = 0.09484
##
## [1] "Checking VECM(4, 2)"
##
## Breusch-Godfrey LM test
##
```

```
## data: Residuals of VAR object VECM_est[[adq_idx_vecm[i]]]
## Chi-squared = 30.057, df = 16, p-value = 0.01771
##
## [1] "Checking VECM(4, 3)"
##
## Breusch-Godfrey LM test
##
## data: Residuals of VAR object VECM_est[[adq_idx_vecm[i]]]
## Chi-squared = 30.153, df = 16, p-value = 0.01723
##
## [1] "Checking VECM(4, 4)"
##
## Breusch-Godfrey LM test
##
## data: Residuals of VAR object VECM_est[[adq_idx_vecm[i]]]
## Chi-squared = 30.39, df = 16, p-value = 0.01608
```

There is nothing surprising about these results. We proceed with the existing specifications as the adequate set, but cautious of the lower p -values produced by the Ljung-Box tests of white noise for the residuals of VECM(4, r) models.

Assignment Project Exam Help

<https://tutorcs.com>

4. Choose a reasonable lag length p^* , and using a subset of specifications (with $p = p^*$) included in the adequate set of VECM models constructed in Question 3, obtain inference on all possible equilibrium relationships. Please use the identifying restrictions $\beta = (\mathbf{I}_r, \beta)$ where β is $(n - r) \times r$.

WeChat: cstutorcs

Solution We will use $p^* = 3$, but the same exercise can be carried out with $p^* = 4$. The `urca` package provides an estimation routine for VECMs called `cajorls`. However, it *does not* provide standard errors for the estimated coefficients in the equilibrium relationships. This is unfortunately a recurring issue in software packages!

We need to construct some measures of uncertainty manually. One way to do this is to test restrictions on elements of the matrix β , one coefficient at a time. The function `blrtest` in the `urca` package implements a different kind of Likelihood Ratio test for this purpose.

```
p <- 3
for (r in 2:4)
{
  if (r < n)
  {
    vec_pr <- ca.jo(x, type = "trace", spec = "transitory")
```

```

beta_pr <- cajorls(vec_pr, r)$beta

bpvals_pr <- beta_pr
bpvals_pr[,] <- NA

for (i in (r + 1):n)
{
  for (j in 1:r)
  {
    H <- beta_pr
    H[i, j] <- 0
    bpvals_pr[i, j] <- blrtest(vec_pr, H, r)$pval[1]
  }
}

print(paste0("Results for VECM(", p, ", ", r, "):"))
print(beta_pr)
print(bpvals_pr)
}

```

Assignment Project Exam Help

```

## [1] "Results for VECM(3, 2):"
##          ect1      ect2
## i90d.l1  1.000000e+00  0.000000
## i180d.l1 -2.220446e-16  1.000000
## i3y.l1   -4.897845e-00 -4.700581
## i5y.l1    4.066957e+00  3.861175
##          ect1      ect2
## i90d.l1          NA          NA
## i180d.l1          NA          NA
## i3y.l1   1.888033e-08  2.599778e-07
## i5y.l1   1.409187e-08  1.096616e-07
## [1] "Results for VECM(3, 3):"
##          ect1      ect2      ect3
## i90d.l1  1.00000000 -4.440892e-16 -1.110223e-16
## i180d.l1  0.00000000  1.000000e+00  0.000000e+00
## i3y.l1    0.00000000  0.000000e+00  1.000000e+00
## i5y.l1   -0.4547867 -4.784092e-01 -9.232109e-01
##          ect1      ect2      ect3
## i90d.l1          NA          NA          NA
## i180d.l1          NA          NA          NA
## i3y.l1           NA          NA          NA
## i5y.l1   4.975749e-09  3.003364e-07  3.1461e-09

```

We begin with the VECM specified by $r = 2$. This is the case where we have two stochastic trends, and by the Granger representation theorem, at least *two*

variables MUST be $I(1)$. The question is: which of $\{i90d_t\}$, $\{i180d_t\}$, $\{i3y_t\}$ or $\{i5y_t\}$ can be regarded as $I(1)$ processes in this context?

Recall from Tutorial 6 that we found evidence for $i90d_t \sim I(0)$. If this is the case, then the first equilibrium relationship, labelled “ect1”, suggests that $\{i90d_t\}$, $\{i3y_t\}$ and $\{i5y_t\}$ are related in equilibrium. Consequently, either $\{i3y_t\}$ and $\{i5y_t\}$ are both $I(0)$ processes, *or* they are both $I(1)$ *and* cointegrated processes.

If both were $I(0)$, then we would have three $I(0)$ variables, which contradicts the fact that we must have at least two $I(1)$ variables. Hence, the only viable implication is that $\{i3y_t\}$ and $\{i5y_t\}$ are cointegrated processes.

The second equilibrium relationship, labelled “ect2”, suggests that $\{i180d_t\}$, $\{i3y_t\}$ and $\{i5y_t\}$ are related in equilibrium. Since $\{i3y_t\}$ and $\{i5y_t\}$ must be cointegrated, $\{i180d_t\}$ *cannot* be $I(0)$ —if that was true, then there would be only *one* common stochastic trend, which again contradicts the fact that we must have *two* stochastic trends.

This leaves us with the possibility that $\{i180d_t\}$, $\{i3y_t\}$ and $\{i5y_t\}$ are all $I(1)$ and cointegrated. However, in this case we have inferred *two* linearly independent cointegrating relations, which again means that there is only *one* stochastic trend present.

We see that if we start with $i90d_t \sim I(0)$ and $r = 2$, then we always arrive at a contradiction! This means that our inference from the VECM with $r = 2$ leads us to conclude that $i90d_t \sim I(1)$, which is clearly not compatible with the inference obtained from ADF tests that lead to concluding $i90d_t \sim I(0)$ with a high degree of confidence!

Turning next to the VECM with $r = 3$, recall that this is the case where we have one stochastic trend, and by the Granger representation theorem, at least one variable *must* be $I(1)$. Again, if we assume $i90d_t \sim I(0)$, then from equilibrium relationship “ect1”, we infer that $i5y_t \sim I(0)$ with a high degree of confidence. This *does not* contradict our inference from ADF tests regarding $\{i5y_t\}$, where we could not obtain conclusive results.

Following on to equilibrium relationship “ect2”, we see that $\{i180d_t\}$ and $\{i5y_t\}$ must also be in equilibrium, and hence, if $\{i5y_t\}$ is an $I(0)$ process so must be $\{i180d_t\}$. Then, because there must exist at least one process that is $I(1)$, this process *must* be $\{i3y_t\}$. Unfortunately, equilibrium relationship “ect3” leads us to conclude that there exists an equilibrium relationship between $\{i3y_t\}$ and $\{i5y_t\}$ with a high degree of confidence. This is again a contradiction since there cannot exist an equilibrium between an $I(0)$ process and an $I(1)$ process!

It looks like imposing rank restrictions on the VAR leads us to inference that is *not* compatible with the inference we obtained from ADF tests. This does not mean that restricted VARs are wrong, and it does not mean that the inference from ADF tests was wrong either! It only highlights the fact these tools, even when used correctly, can lead to contradicting conclusions.

This is something we must always be aware of. The way to resolve these conflicts is to focus on the ultimate objective. If the goal is to decompose each individual process into trend and cyclical components, then we would rely more on the ADF tests and less on VECMs. If, on the other hand, the goal is to obtain inference on dynamic responses, inference from ADF tests on individual processes is less important, and the adequate set we have constructed with VECMs provides the necessary empirical tools.

5. Compute impulse response functions for 5-year horizons using a Cholesky decomposition for all the specifications in the adequate set of VECM models. Compare these IRFs to those obtained from unrestricted VAR models. Comment on your findings.
-

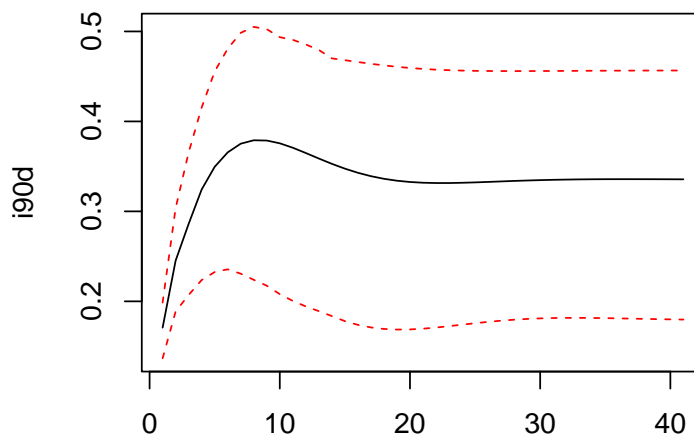
Solution Generating IRFs from estimated VECMs can be done with exactly the same steps, once the `vec2var` function has been used to obtain a VAR representation from restricted VECMs (with n variables).

```
vnames <- c("i90d", "i180d", "i3y", "i5y")
nmods <- length(adq_idx_vecm)
for (i in 1:4)
{
  for (j in 1:4)
  {
    for (imod in 1:nmods)
    {
      p <- adq_set_vecm[imod, 1]
      r <- adq_set_vecm[imod, 2]
      title_i_j <- paste0("VECM(", p, ", ", r,
                          "): Response of ", vnames[i],
                          " to a shock in ", vnames[j])

      irf_i_j <- irf(VECM_est[[adq_idx_vecm[imod]]],
                     n.ahead = 40,
                     response = vnames[i],
                     impulse = vnames[j],
                     boot = TRUE)

      plot(irf_i_j, main = title_i_j)
      cat("\r", title_i_j, " ", sep = "")
    }
  }
}
```

VECM(3, 2): Response of i90d to a shock in i90d

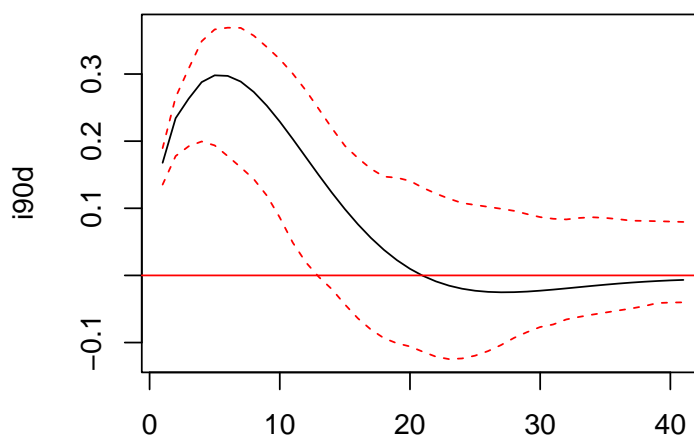


Assignment Project Exam Help

VECM(3, 2): Response of i90d to a shock in i90d

<https://tutorcs.com>

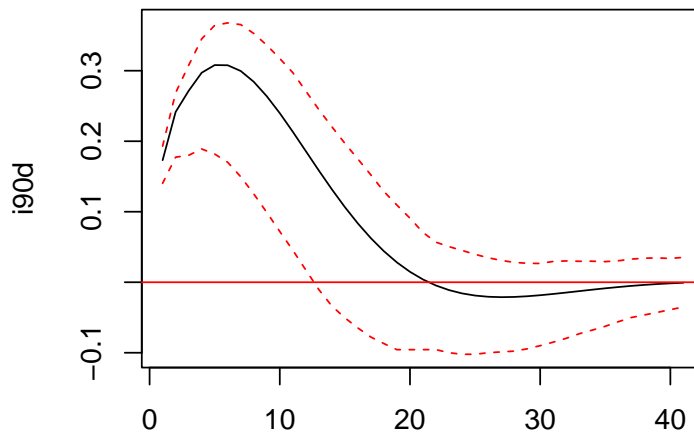
WeChat: cstutorcs



95 % Bootstrap CI, 100 runs

VECM(3, 3): Response of i90d to a shock in i90d

VECM(3, 4): Response of i90d to a shock in i90d

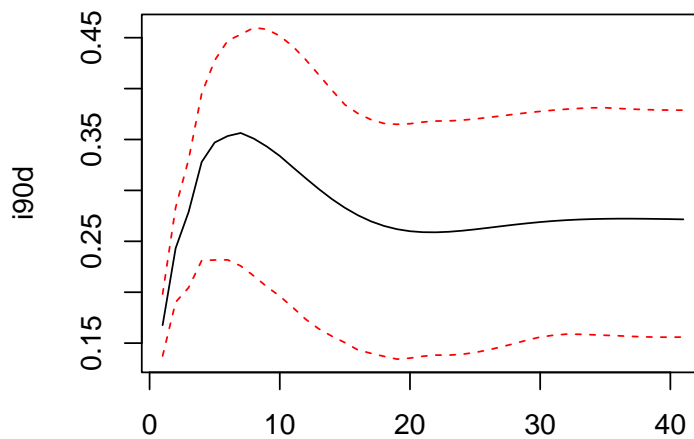


95 % Bootstrap CI, 100 runs
Assignment Project Exam Help

VECM(3, 4): Response of i90d to a shock in i90d

<https://tutorcs.com>

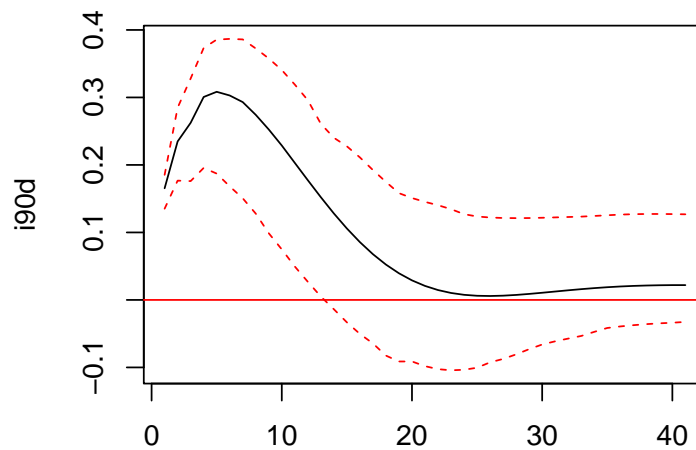
VECM(4, 2): Response of i90d to a shock in i90d
WeChat: cstutorcs



95 % Bootstrap CI, 100 runs

VECM(4, 2): Response of i90d to a shock in i90d

VECM(4, 3): Response of i90d to a shock in i90d



Assignment Project Exam Help

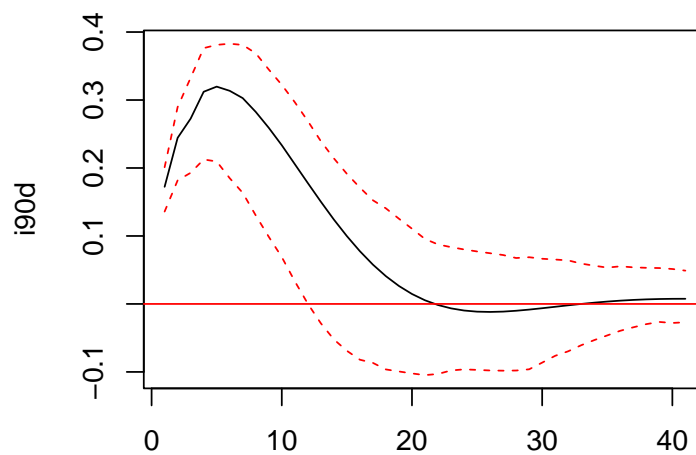
95 % Bootstrap CI, 100 runs

VECM(4, 3): Response of i90d to a shock in i90d

<https://tutorcs.com>

WeChat: cstutorcs

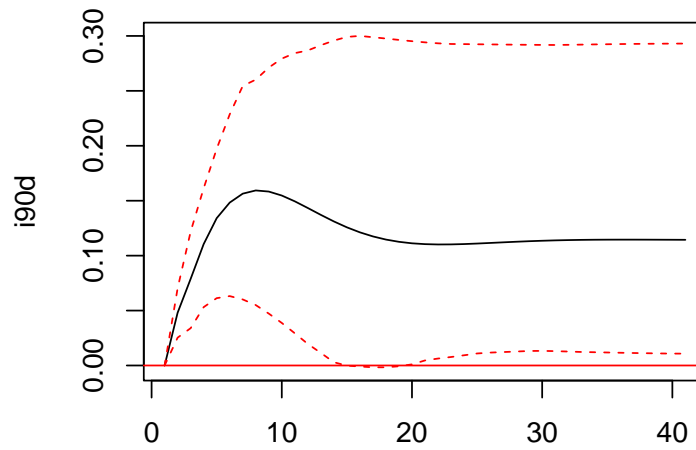
VECM(4, 4): Response of i90d to a shock in i90d



95 % Bootstrap CI, 100 runs

VECM(4, 4): Response of i90d to a shock in i90d

VECM(3, 2): Response of i90d to a shock in i180d

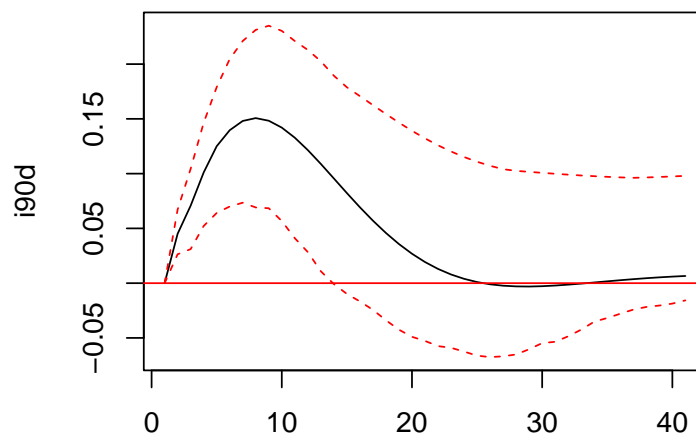


Assignment Project Exam Help

VECM(3, 2): Response of i90d to a shock in i180d

<https://tutorcs.com>

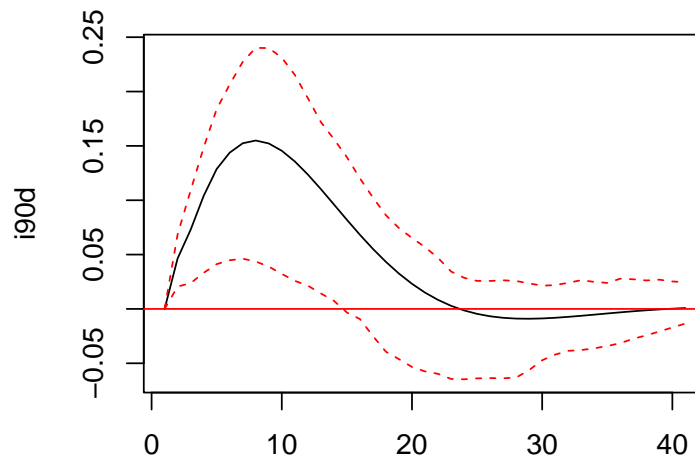
WeChat: cstutorcs



95 % Bootstrap CI, 100 runs

VECM(3, 3): Response of i90d to a shock in i180d

VECM(3, 4): Response of i90d to a shock in i180d



95 % Bootstrap CI, 100 runs

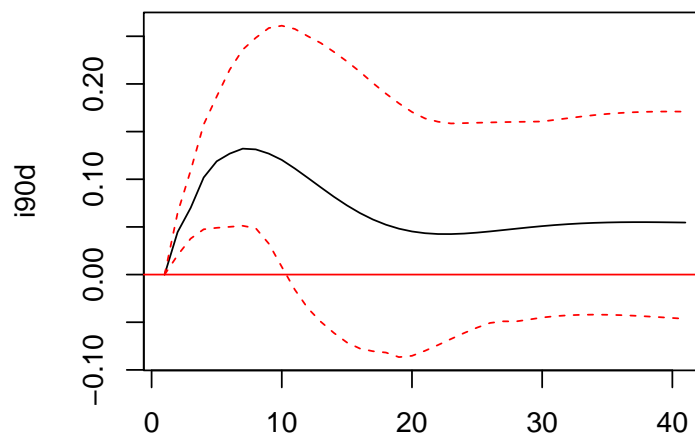
Assignment Project Exam Help

VECM(3, 4): Response of i90d to a shock in i180d

<https://tutorcs.com>

VECM(4, 2): Response of i90d to a shock in i180d

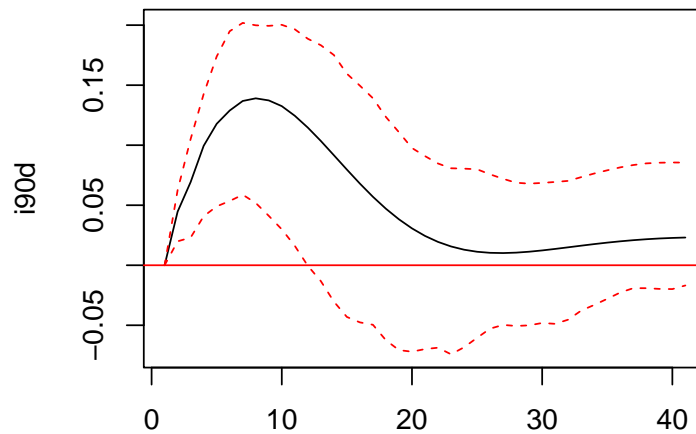
WeChat: cstutorcs



95 % Bootstrap CI, 100 runs

VECM(4, 2): Response of i90d to a shock in i180d

VECM(4, 3): Response of i90d to a shock in i180d

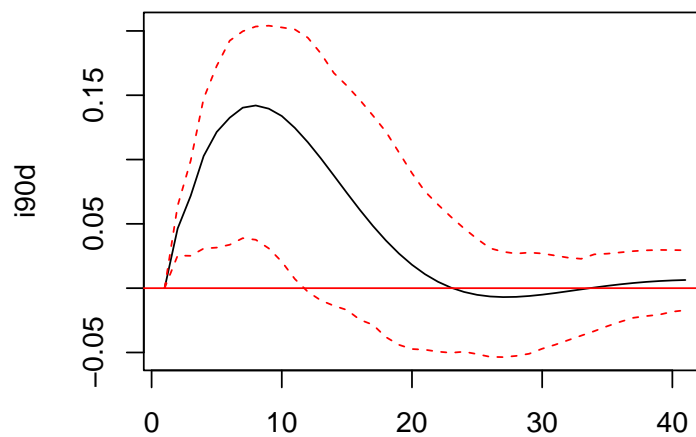


95 % Bootstrap CI, 100 runs
Assignment Project Exam Help

VECM(4, 3): Response of i90d to a shock in i180d

<https://tutorcs.com>

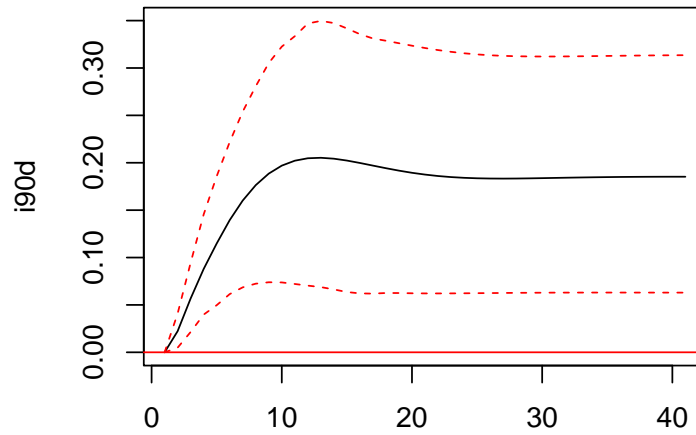
VECM(4, 4): Response of i90d to a shock in i180d
WeChat: cstutorcs



95 % Bootstrap CI, 100 runs

VECM(4, 4): Response of i90d to a shock in i180d

VECM(3, 2): Response of i90d to a shock in i3y



Assignment Project Exam Help

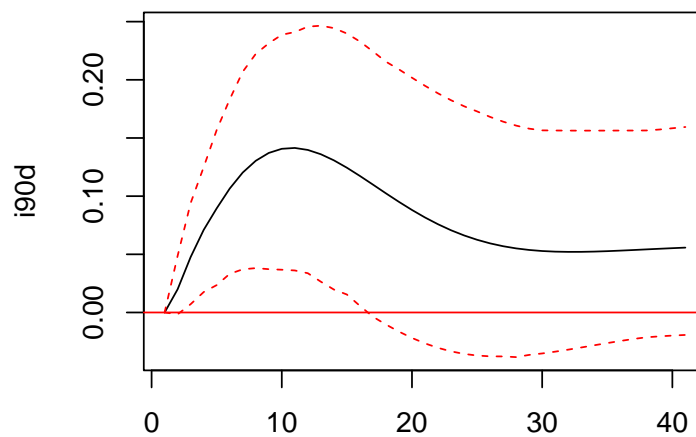
95 % Bootstrap CI, 100 runs

VECM(3, 2): Response of i90d to a shock in i3y

<https://tutorcs.com>

WeChat: cstutorcs

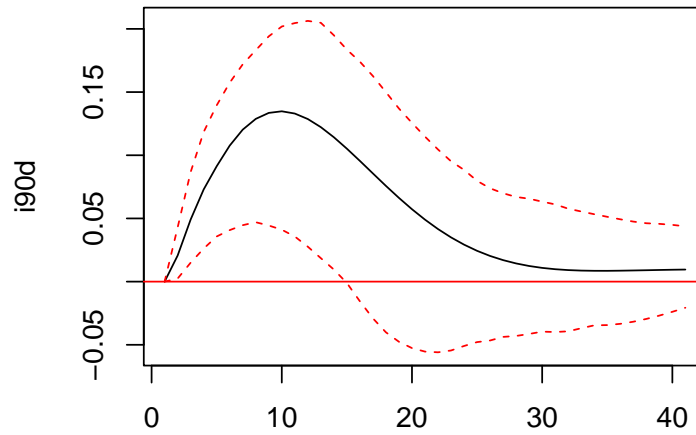
VECM(3, 3): Response of i90d to a shock in i3y



95 % Bootstrap CI, 100 runs

VECM(3, 3): Response of i90d to a shock in i3y

VECM(3, 4): Response of i90d to a shock in i3y



Assignment Project Exam Help

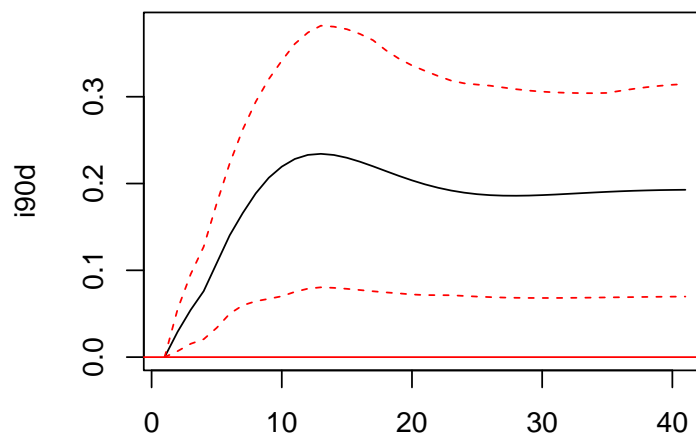
95 % Bootstrap CI, 100 runs

VECM(3, 4): Response of i90d to a shock in i3y

<https://tutorcs.com>

WeChat: cstutorcs

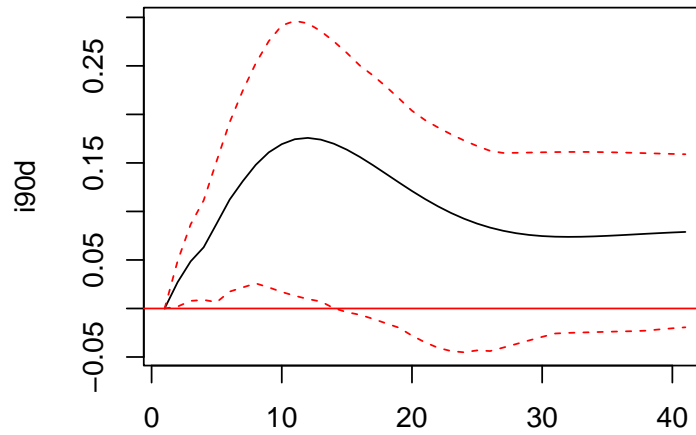
VECM(4, 2): Response of i90d to a shock in i3y



95 % Bootstrap CI, 100 runs

VECM(4, 2): Response of i90d to a shock in i3y

VECM(4, 3): Response of i90d to a shock in i3y



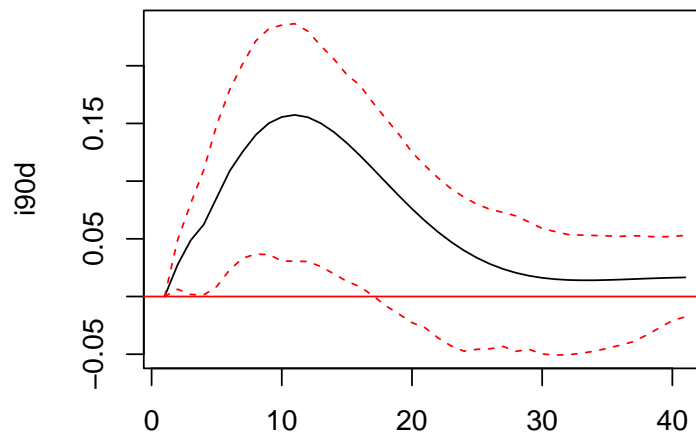
Assignment Project Exam Help

VECM(4, 3): Response of i90d to a shock in i3y

<https://tutorcs.com>

WeChat: cstutorcs

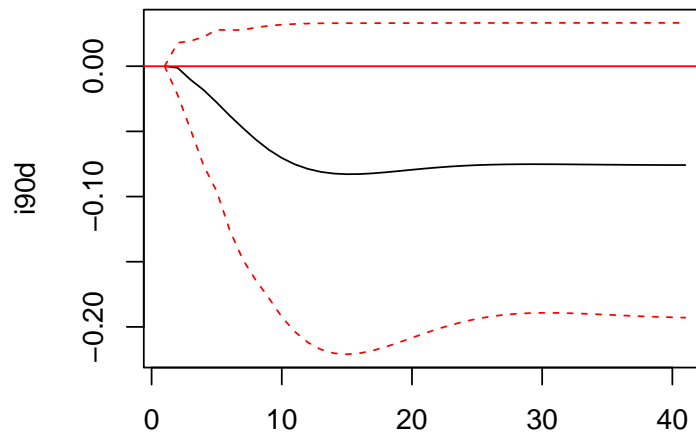
VECM(4, 4): Response of i90d to a shock in i3y



95 % Bootstrap CI, 100 runs

VECM(4, 4): Response of i90d to a shock in i3y

VECM(3, 2): Response of i90d to a shock in i5y

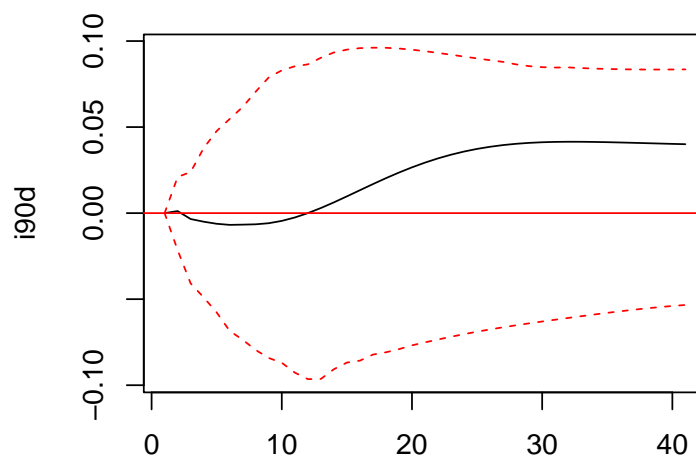


Assignment Project Exam Help

VECM(3, 2): Response of i90d to a shock in i5y

<https://tutorcs.com>

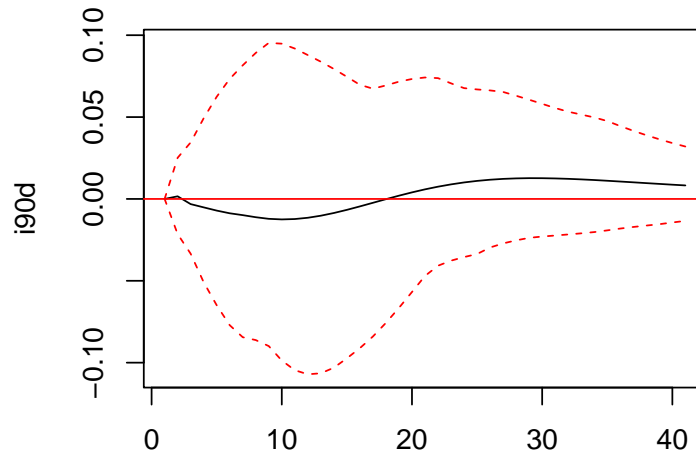
WeChat: cstutorcs



95 % Bootstrap CI, 100 runs

VECM(3, 3): Response of i90d to a shock in i5y

VECM(3, 4): Response of i90d to a shock in i5y



Assignment Project Exam Help

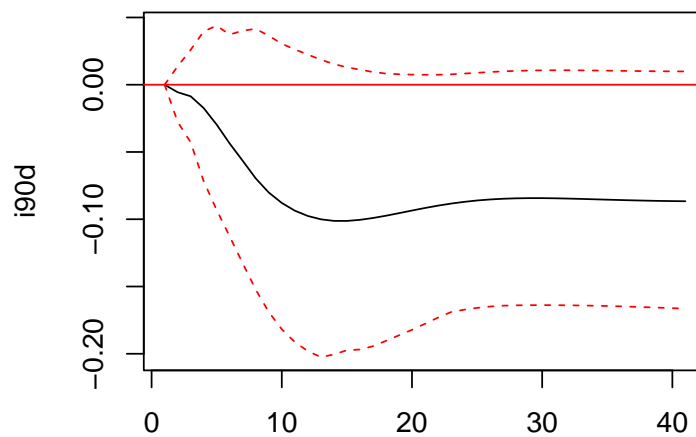
95 % Bootstrap CI, 100 runs

VECM(3, 4): Response of i90d to a shock in i5y

<https://tutorcs.com>

WeChat: cstutorcs

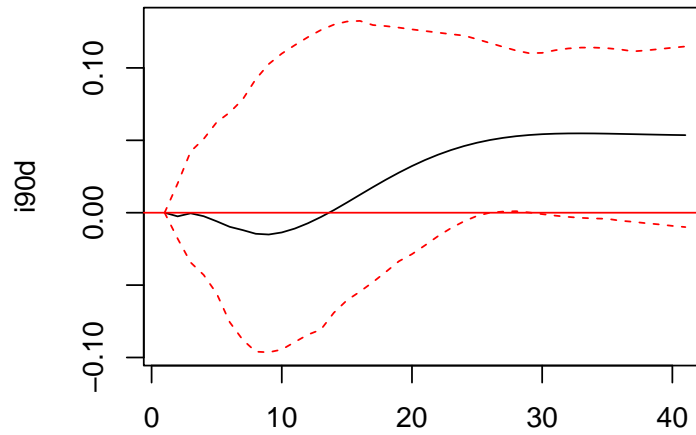
VECM(4, 2): Response of i90d to a shock in i5y



95 % Bootstrap CI, 100 runs

VECM(4, 2): Response of i90d to a shock in i5y

VECM(4, 3): Response of i90d to a shock in i5y

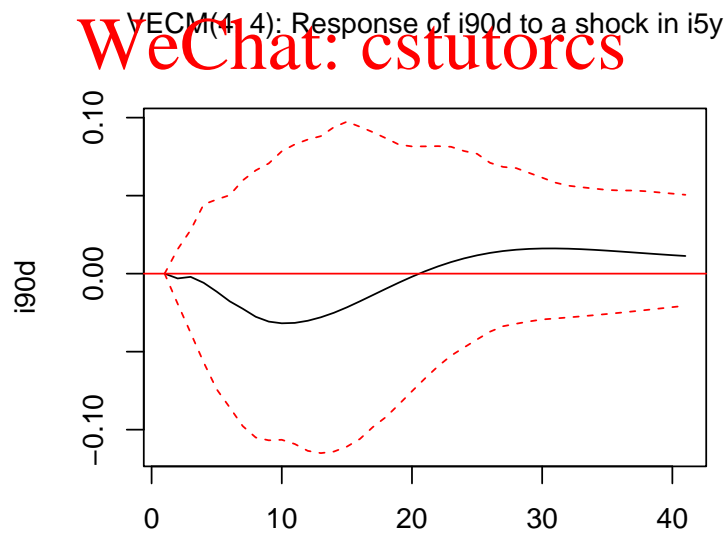


Assignment Project Exam Help

VECM(4, 3): Response of i90d to a shock in i5y

<https://tutorcs.com>

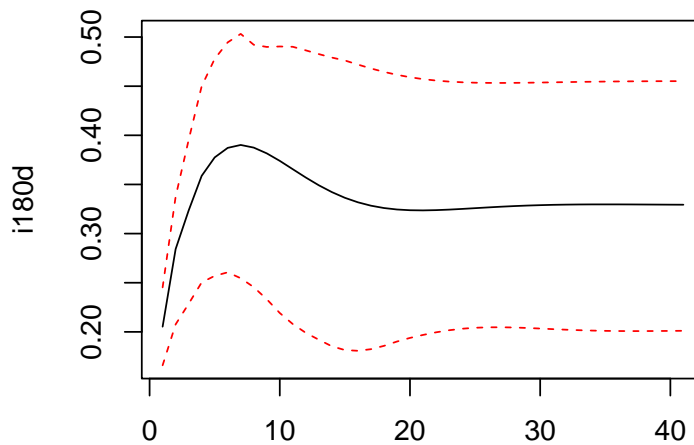
WeChat: cstutorcs



95 % Bootstrap CI, 100 runs

VECM(4, 4): Response of i90d to a shock in i5y

VECM(3, 2): Response of i180d to a shock in i90d

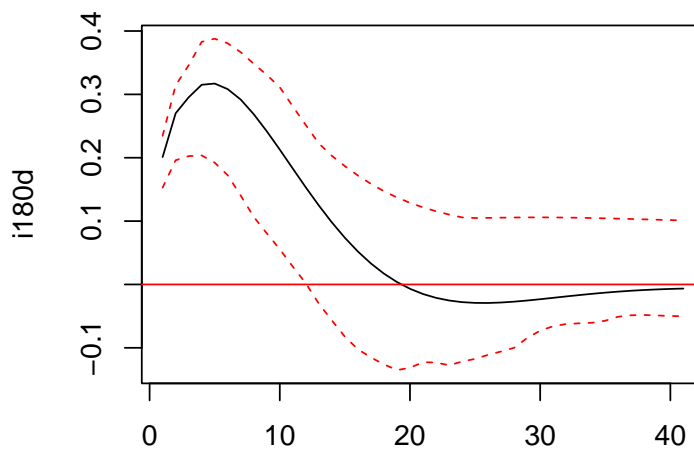


Assignment Project Exam Help

VECM(3, 2): Response of i180d to a shock in i90d

<https://tutorcs.com>

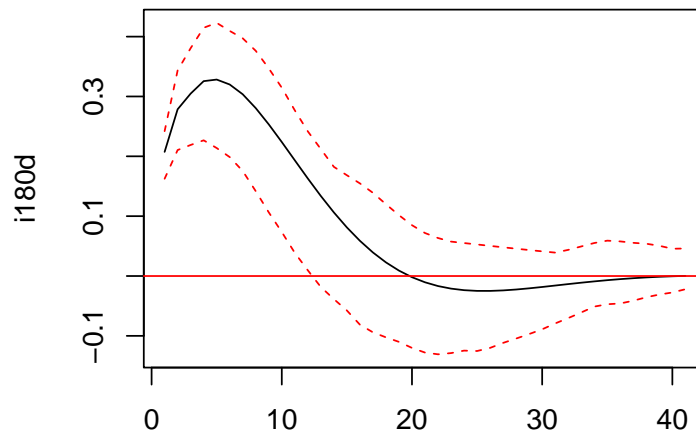
WeChat: cstutorcs



95 % Bootstrap CI, 100 runs

VECM(3, 3): Response of i180d to a shock in i90d

VECM(3, 4): Response of i180d to a shock in i90d



95 % Bootstrap CI, 100 runs

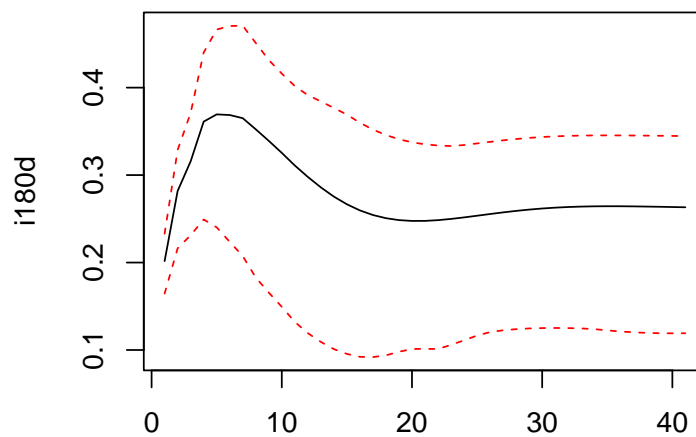
Assignment Project Exam Help

VECM(3, 4): Response of i180d to a shock in i90d

<https://tutorcs.com>

VECM(4, 2): Response of i180d to a shock in i90d

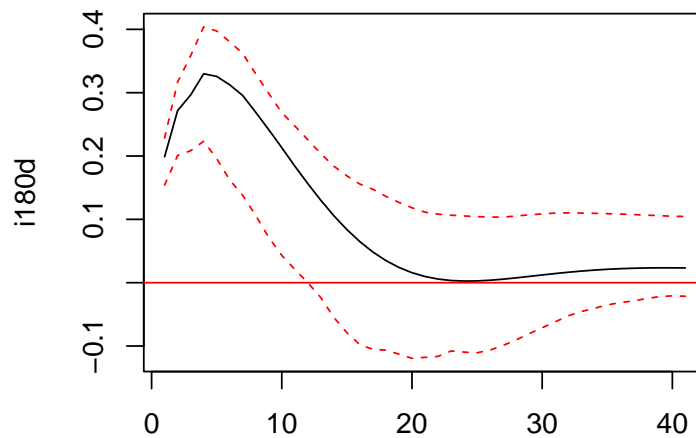
WeChat: cstutorcs



95 % Bootstrap CI, 100 runs

VECM(4, 2): Response of i180d to a shock in i90d

VECM(4, 3): Response of i180d to a shock in i90d



Assignment Project Exam Help

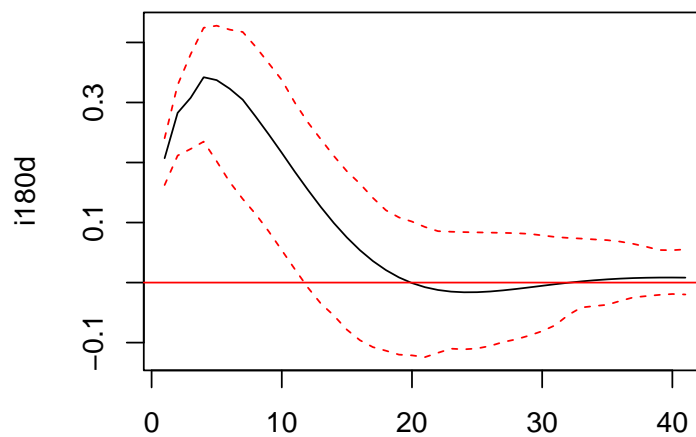
95 % Bootstrap CI, 100 runs

VECM(4, 3): Response of i180d to a shock in i90d

<https://tutorcs.com>

VECM(4, 4): Response of i180d to a shock in i90d

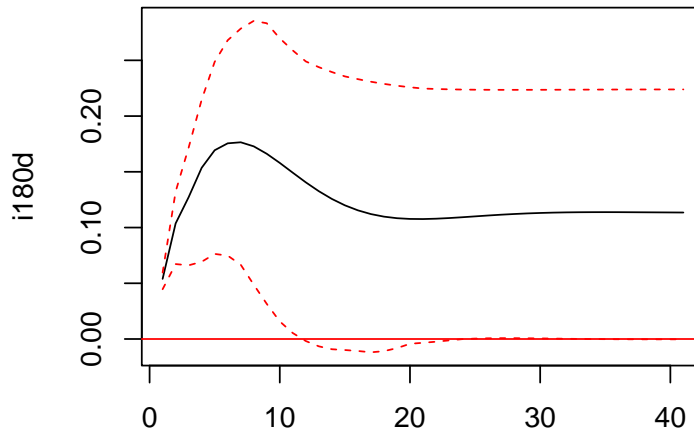
WeChat: cstutorcs



95 % Bootstrap CI, 100 runs

VECM(4, 4): Response of i180d to a shock in i90d

VECM(3, 2): Response of i180d to a shock in i180d

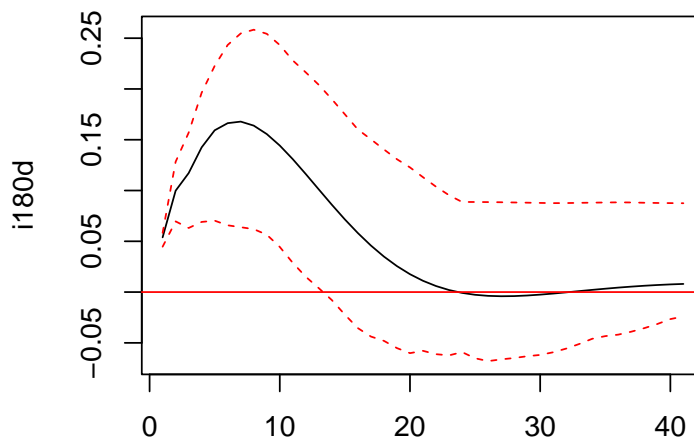


Assignment Project Exam Help

VECM(3, 2): Response of i180d to a shock in i180d

<https://tutorcs.com>

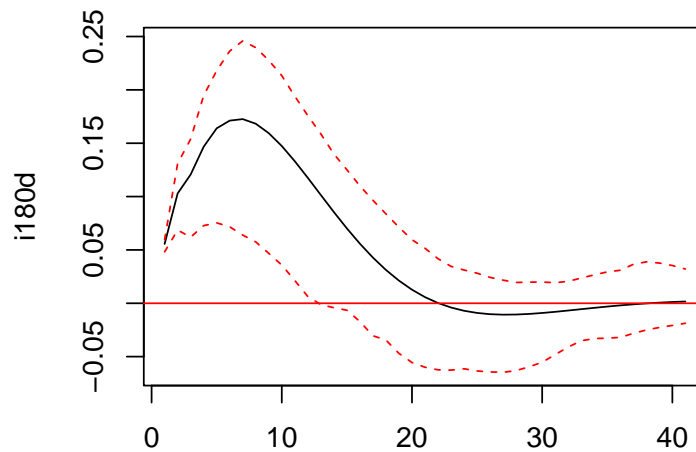
VECM(3, 3): Response of i180d to a shock in i180d



95 % Bootstrap CI, 100 runs

VECM(3, 3): Response of i180d to a shock in i180d

VECM(3, 4): Response of i180d to a shock in i180d

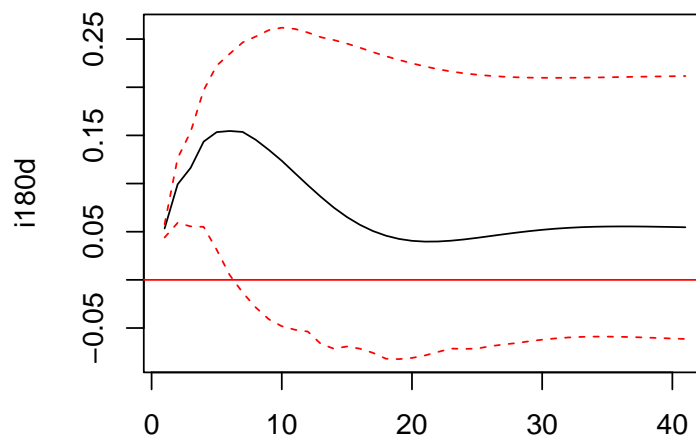


95 % Bootstrap CI, 100 runs
Assignment Project Exam Help

VECM(3, 4): Response of i180d to a shock in i180d

<https://tutorcs.com>

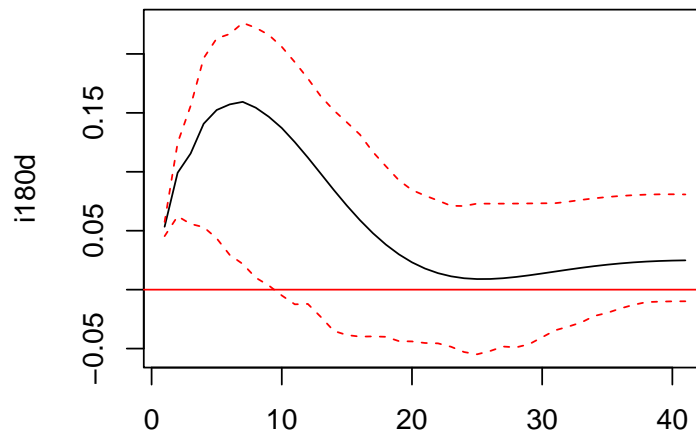
VECM(4, 2): Response of i180d to a shock in i180d
WeChat: cstutorcs



95 % Bootstrap CI, 100 runs

VECM(4, 2): Response of i180d to a shock in i180d

VECM(4, 3): Response of i180d to a shock in i180d

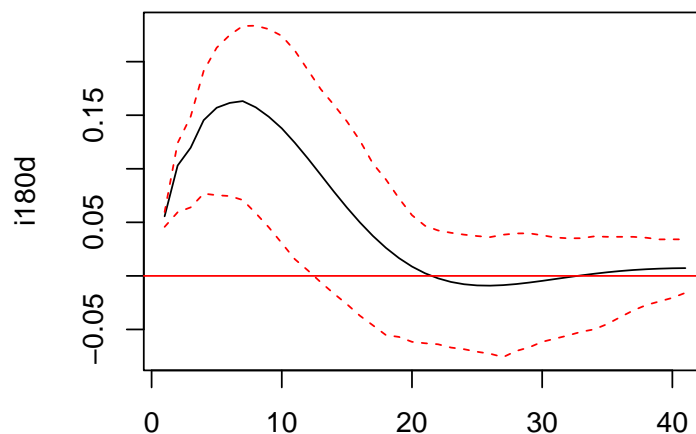


95 % Bootstrap CI, 100 runs
Assignment Project Exam Help

VECM(4, 3): Response of i180d to a shock in i180d

<https://tutorcs.com>

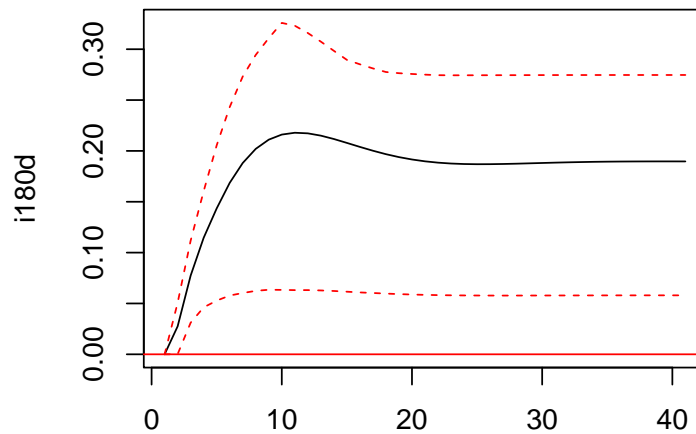
VECM(4, 4): Response of i180d to a shock in i180d
WeChat: cstutorcs



95 % Bootstrap CI, 100 runs

VECM(4, 4): Response of i180d to a shock in i180d

VECM(3, 2): Response of i180d to a shock in i3y

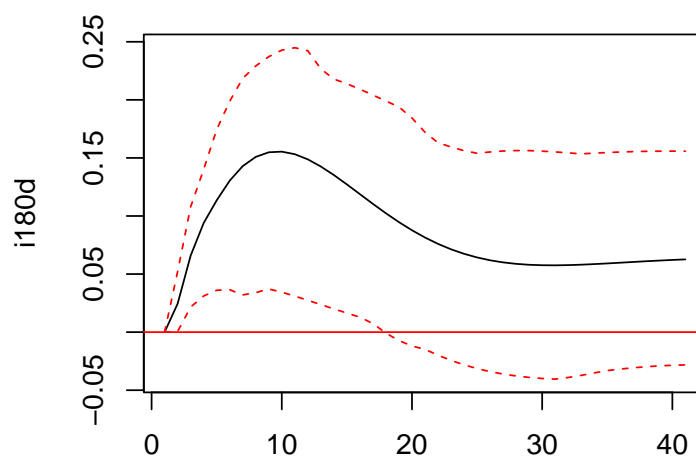


Assignment Project Exam Help

VECM(3, 2): Response of i180d to a shock in i3y

<https://tutorcs.com>

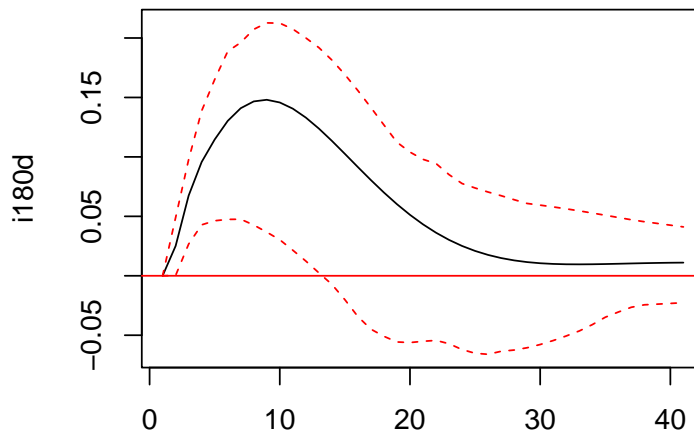
WeChat: cstutorcs



95 % Bootstrap CI, 100 runs

VECM(3, 3): Response of i180d to a shock in i3y

VECM(3, 4): Response of i180d to a shock in i3y

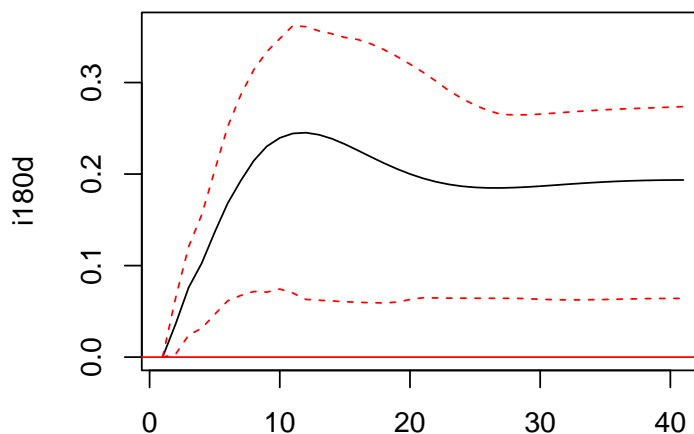


95 % Bootstrap CI, 100 runs
Assignment Project Exam Help

VECM(3, 4): Response of i180d to a shock in i3y

<https://tutorcs.com>

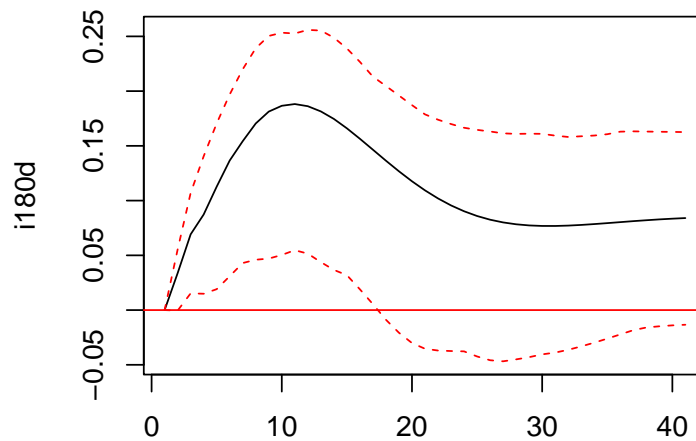
VECM(4, 2): Response of i180d to a shock in i3y
WeChat: cstutorcs



95 % Bootstrap CI, 100 runs

VECM(4, 2): Response of i180d to a shock in i3y

VECM(4, 3): Response of i180d to a shock in i3y



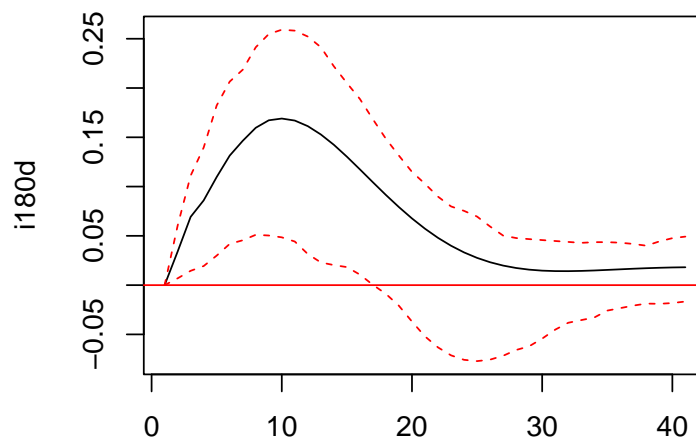
Assignment Project Exam Help

VECM(4, 3): Response of i180d to a shock in i3y

<https://tutorcs.com>

WeChat: cstutorcs

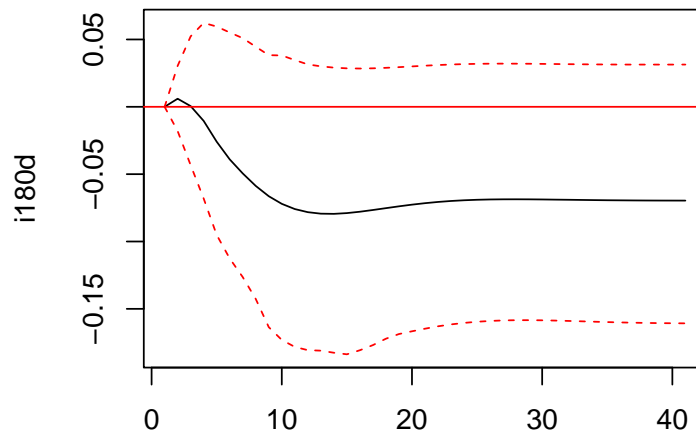
VECM(4, 4): Response of i180d to a shock in i3y



95 % Bootstrap CI, 100 runs

VECM(4, 4): Response of i180d to a shock in i3y

VECM(3, 2): Response of i180d to a shock in i5y

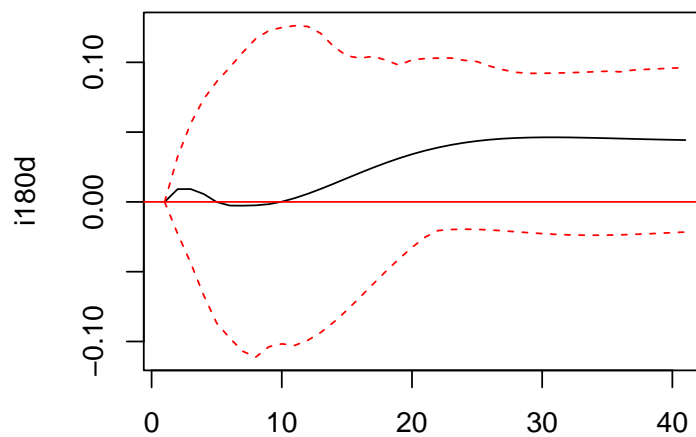


Assignment Project Exam Help

VECM(3, 2): Response of i180d to a shock in i5y

<https://tutorcs.com>

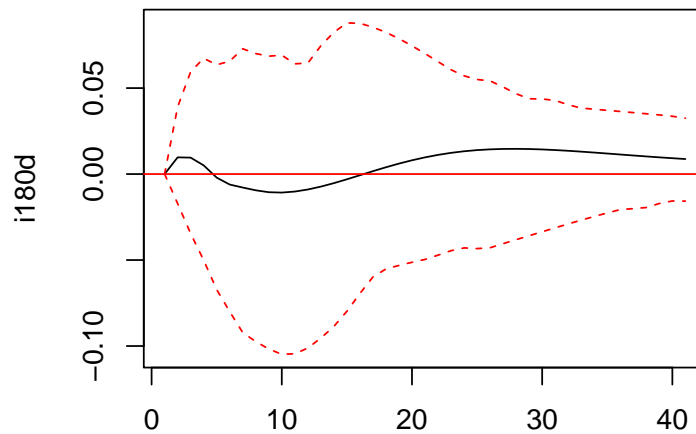
WeChat: cstutorcs



95 % Bootstrap CI, 100 runs

VECM(3, 3): Response of i180d to a shock in i5y

VECM(3, 4): Response of i180d to a shock in i5y



95 % Bootstrap CI, 100 runs

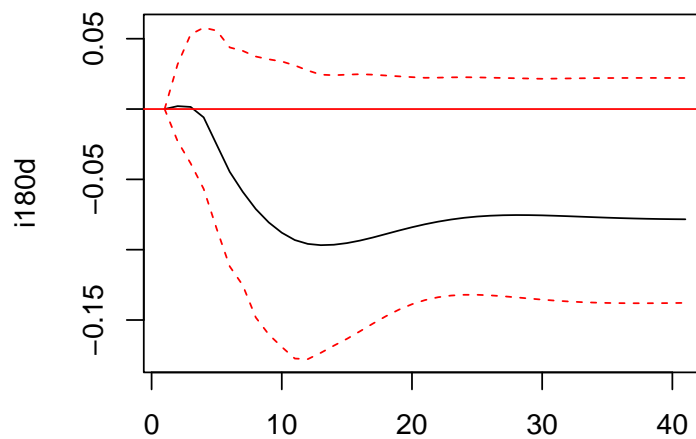
Assignment Project Exam Help

VECM(3, 4): Response of i180d to a shock in i5y

<https://tutorcs.com>

VECM(4, 2): Response of i180d to a shock in i5y

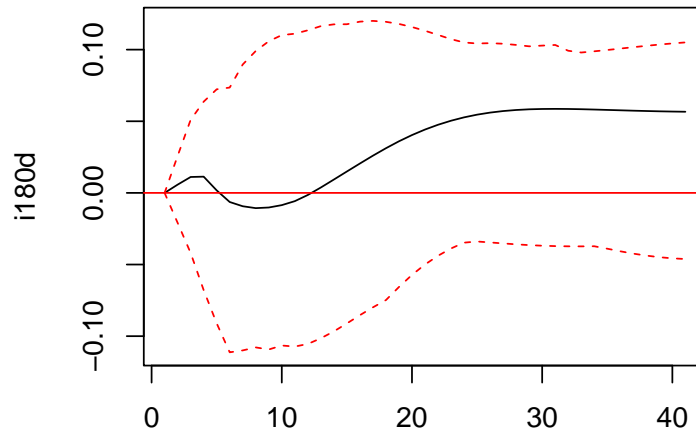
WeChat: cstutorcs



95 % Bootstrap CI, 100 runs

VECM(4, 2): Response of i180d to a shock in i5y

VECM(4, 3): Response of i180d to a shock in i5y



95 % Bootstrap CI, 100 runs

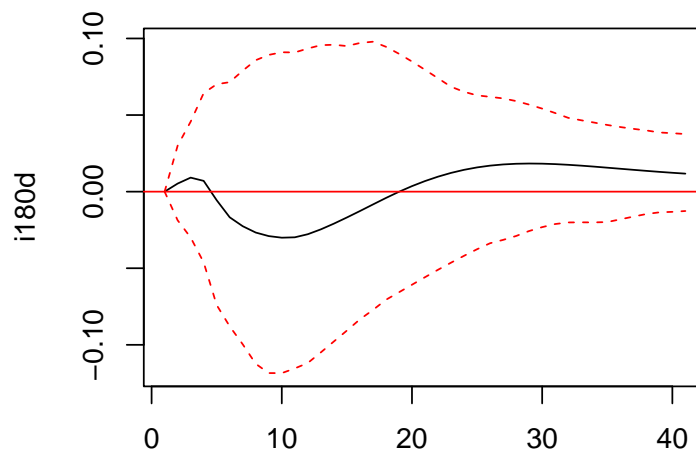
Assignment Project Exam Help

VECM(4, 3): Response of i180d to a shock in i5y

<https://tutorcs.com>

VECM(4, 4): Response of i180d to a shock in i5y

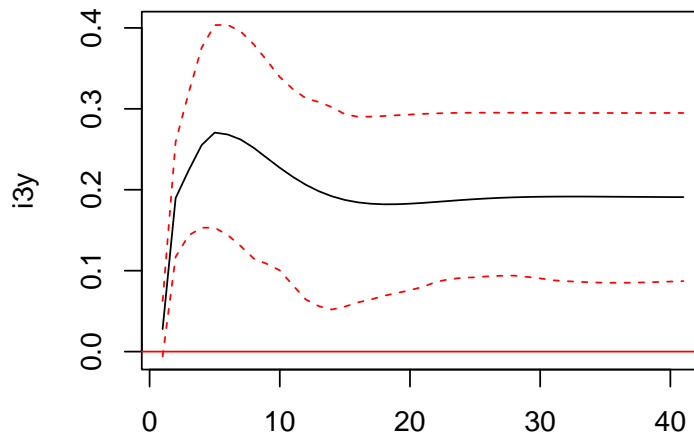
WeChat: cstutorcs



95 % Bootstrap CI, 100 runs

VECM(4, 4): Response of i180d to a shock in i5y

VECM(3, 2): Response of i3y to a shock in i90d



Assignment Project Exam Help

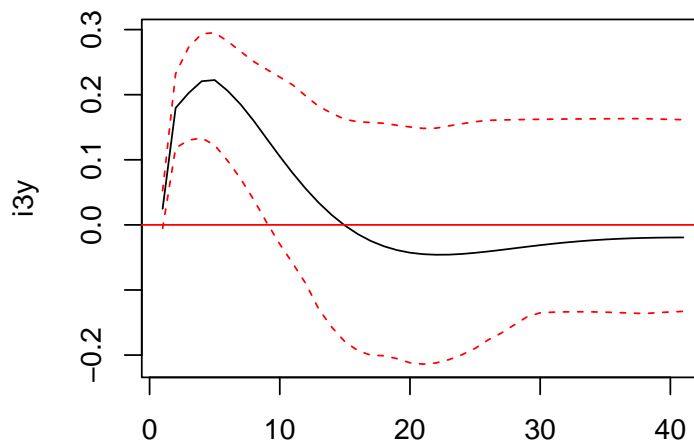
95 % Bootstrap CI, 100 runs

VECM(3, 2): Response of i3y to a shock in i90d

<https://tutorcs.com>

WeChat: cstutorcs

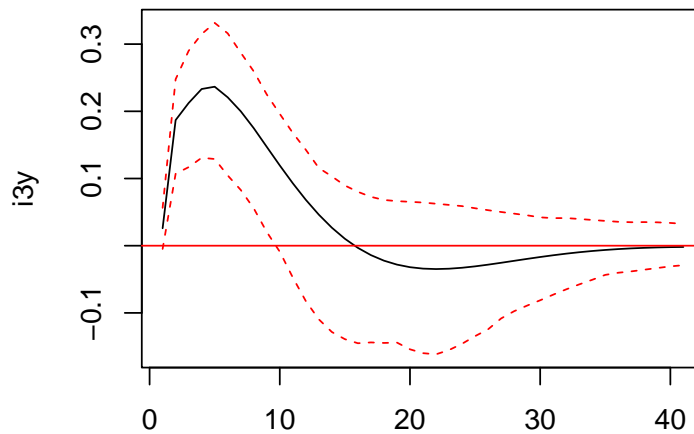
VECM(3, 3): Response of i3y to a shock in i90d



95 % Bootstrap CI, 100 runs

VECM(3, 3): Response of i3y to a shock in i90d

VECM(3, 4): Response of i3y to a shock in i90d

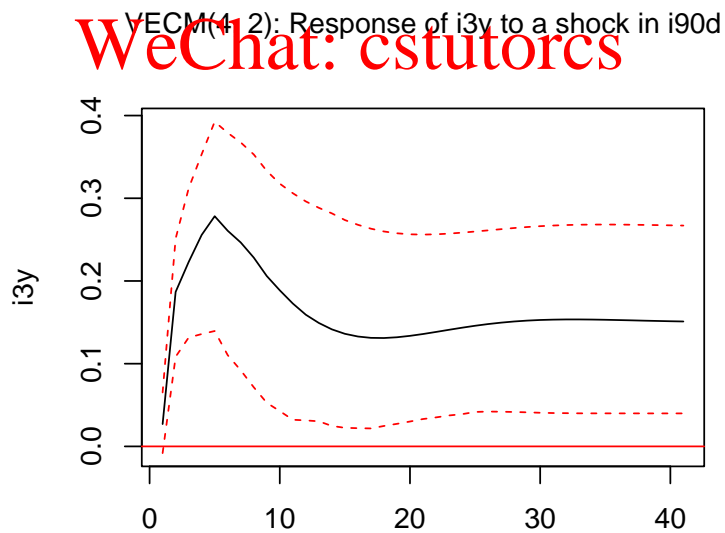


Assignment Project Exam Help

VECM(3, 4): Response of i3y to a shock in i90d

<https://tutorcs.com>

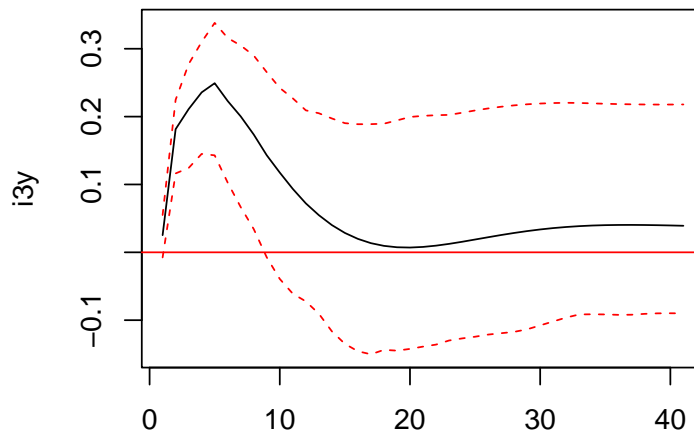
WeChat: cstutorcs



95 % Bootstrap CI, 100 runs

VECM(4, 2): Response of i3y to a shock in i90d

VECM(4, 3): Response of i3y to a shock in i90d



95 % Bootstrap CI, 100 runs

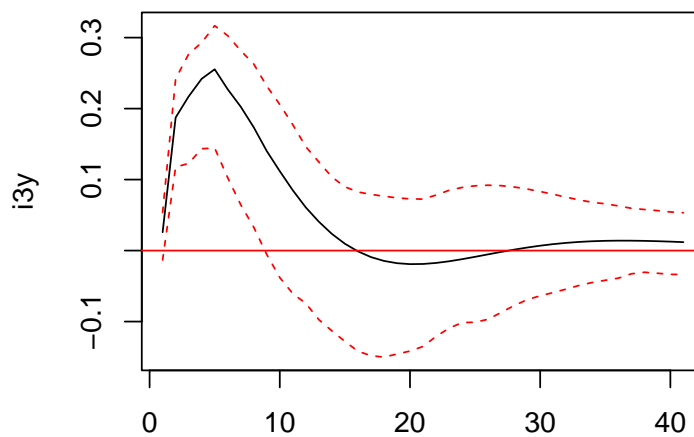
Assignment Project Exam Help

VECM(4, 3): Response of i3y to a shock in i90d

<https://tutorcs.com>

WeChat: cstutorcs

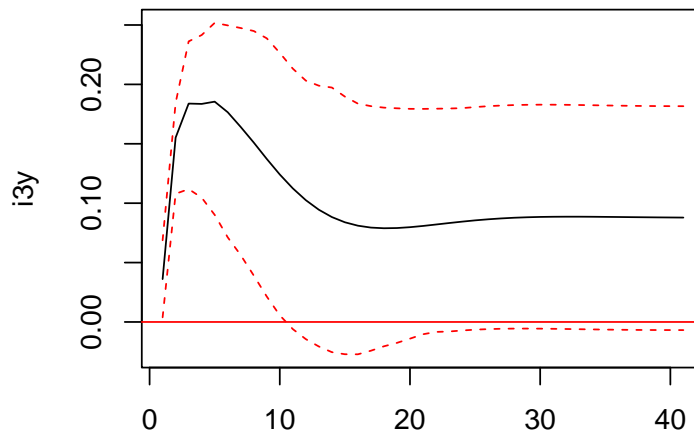
VECM(4, 4): Response of i3y to a shock in i90d



95 % Bootstrap CI, 100 runs

VECM(4, 4): Response of i3y to a shock in i90d

VECM(3, 2): Response of i3y to a shock in i180d

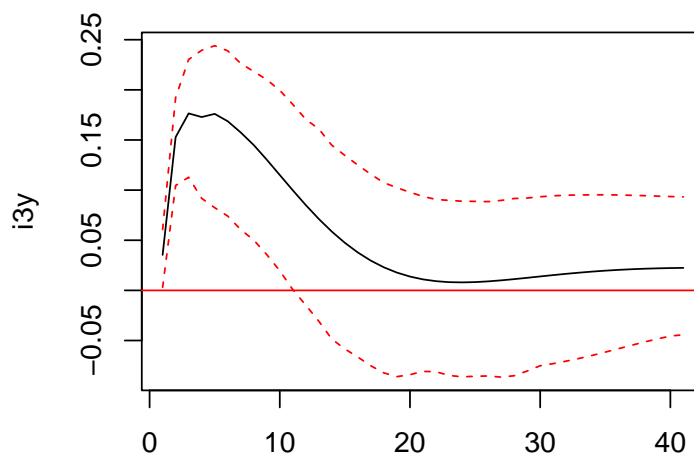


Assignment Project Exam Help

VECM(3, 2): Response of i3y to a shock in i180d

<https://tutorcs.com>

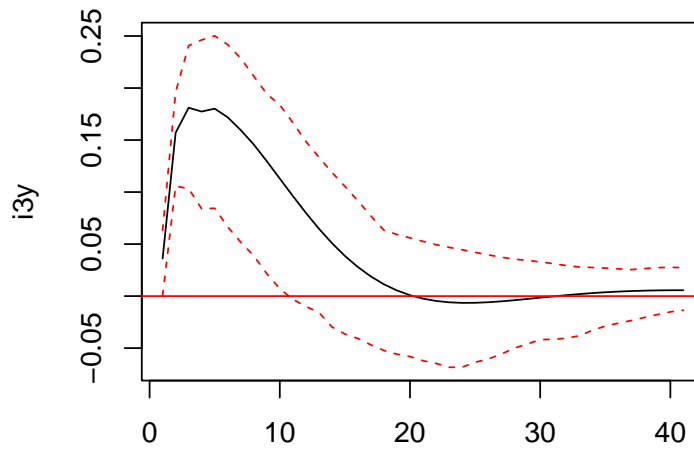
WeChat: cstutorcs



95 % Bootstrap CI, 100 runs

VECM(3, 3): Response of i3y to a shock in i180d

VECM(3, 4): Response of i3y to a shock in i180d



Assignment Project Exam Help

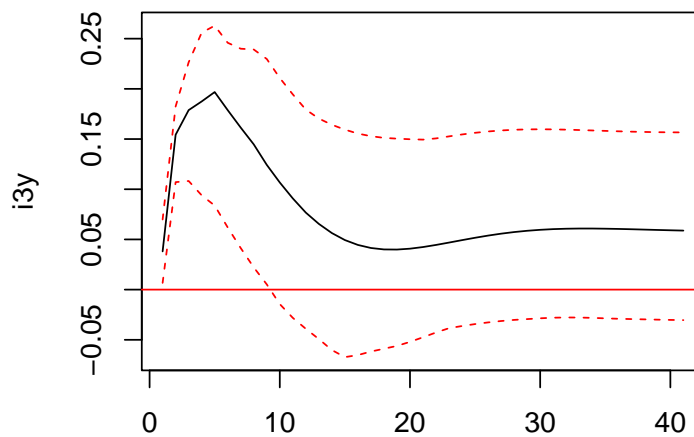
95 % Bootstrap CI, 100 runs

VECM(3, 4): Response of i3y to a shock in i180d

<https://tutorcs.com>

WeChat: cstutorcs

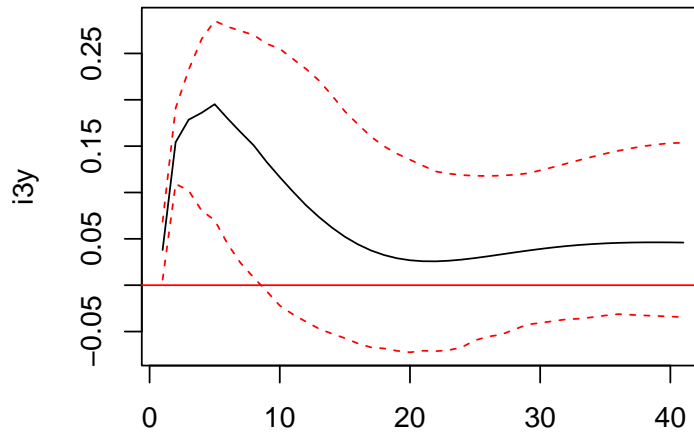
VECM(4, 2): Response of i3y to a shock in i180d



95 % Bootstrap CI, 100 runs

VECM(4, 2): Response of i3y to a shock in i180d

VECM(4, 3): Response of i3y to a shock in i180d



95 % Bootstrap CI, 100 runs

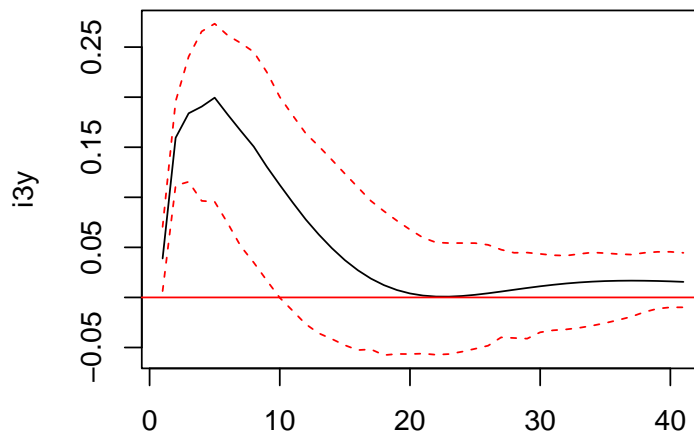
Assignment Project Exam Help

VECM(4, 3): Response of i3y to a shock in i180d

<https://tutorcs.com>

VECM(4, 4): Response of i3y to a shock in i180d

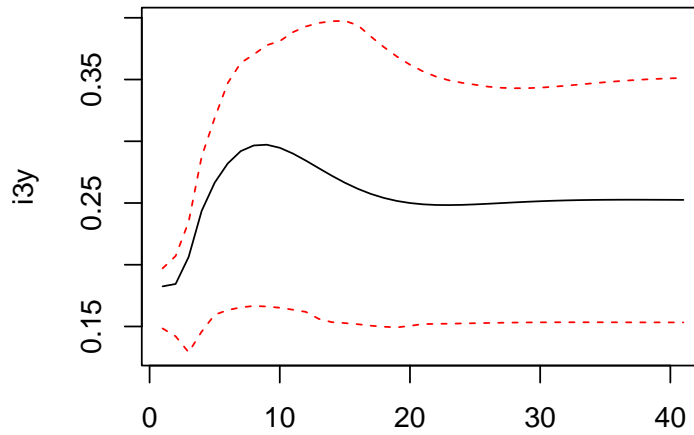
WeChat: cstutorcs



95 % Bootstrap CI, 100 runs

VECM(4, 4): Response of i3y to a shock in i180d

VECM(3, 2): Response of i3y to a shock in i3y



Assignment Project Exam Help

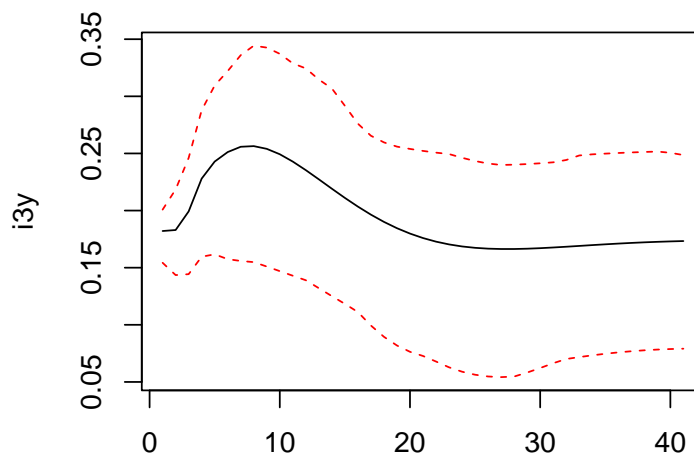
95 % Bootstrap CI, 100 runs

VECM(3, 2): Response of i3y to a shock in i3y

<https://tutorcs.com>

WeChat: cstutorcs

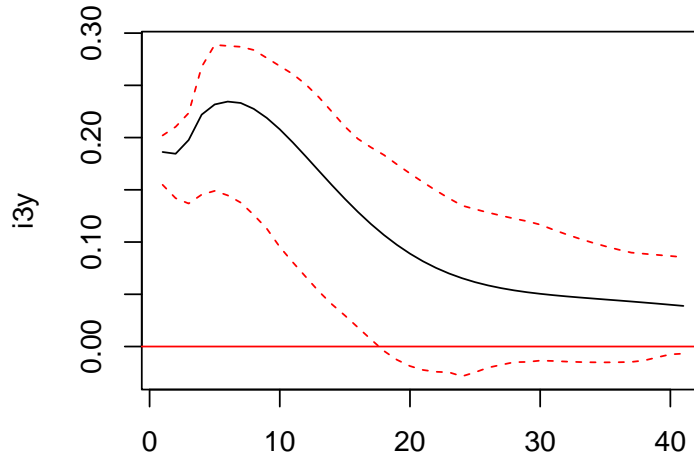
VECM(3, 3): Response of i3y to a shock in i3y



95 % Bootstrap CI, 100 runs

VECM(3, 3): Response of i3y to a shock in i3y

VECM(3, 4): Response of i3y to a shock in i3y



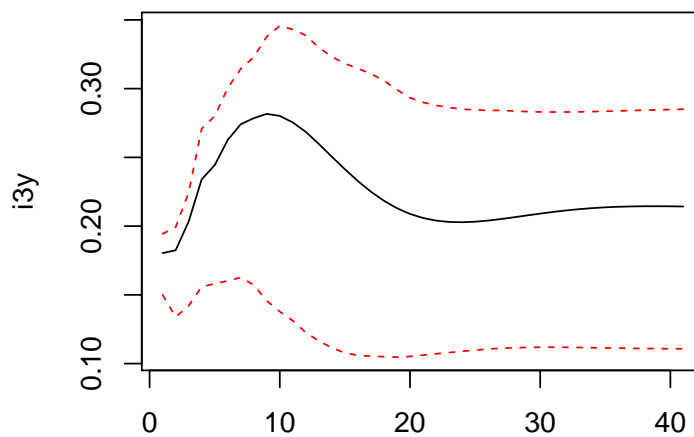
Assignment Project Exam Help

VECM(3, 4): Response of i3y to a shock in i3y

<https://tutorcs.com>

WeChat: cstutorcs

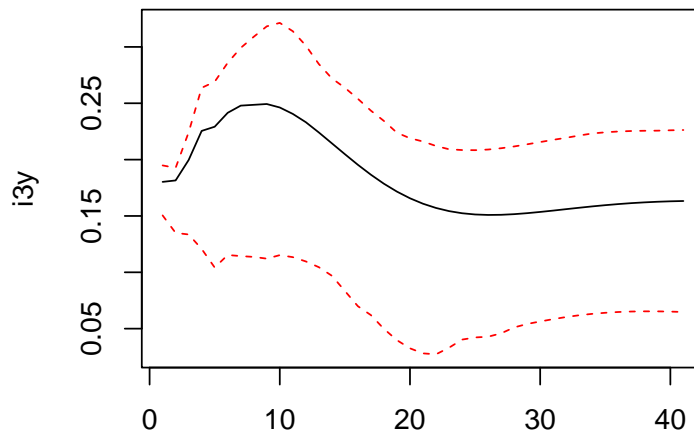
VECM(4, 2): Response of i3y to a shock in i3y



95 % Bootstrap CI, 100 runs

VECM(4, 2): Response of i3y to a shock in i3y

VECM(4, 3): Response of i3y to a shock in i3y



Assignment Project Exam Help

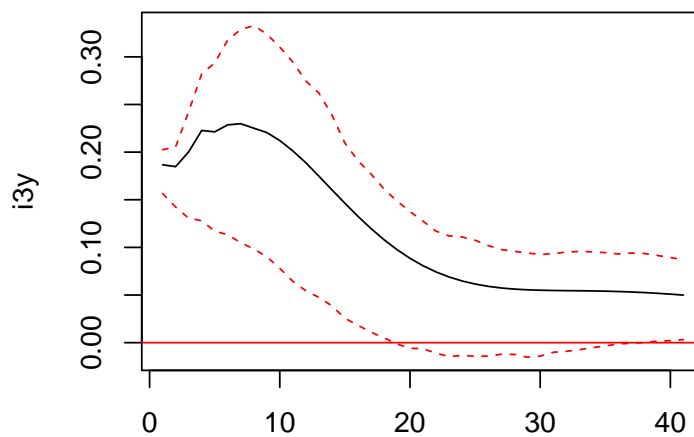
95 % Bootstrap CI, 100 runs

VECM(4, 3): Response of i3y to a shock in i3y

<https://tutorcs.com>

WeChat: cstutorcs

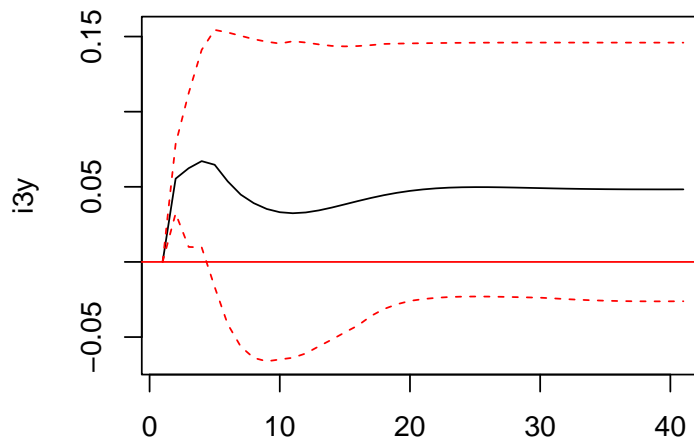
VECM(4, 4): Response of i3y to a shock in i3y



95 % Bootstrap CI, 100 runs

VECM(4, 4): Response of i3y to a shock in i3y

VECM(3, 2): Response of i3y to a shock in i5y



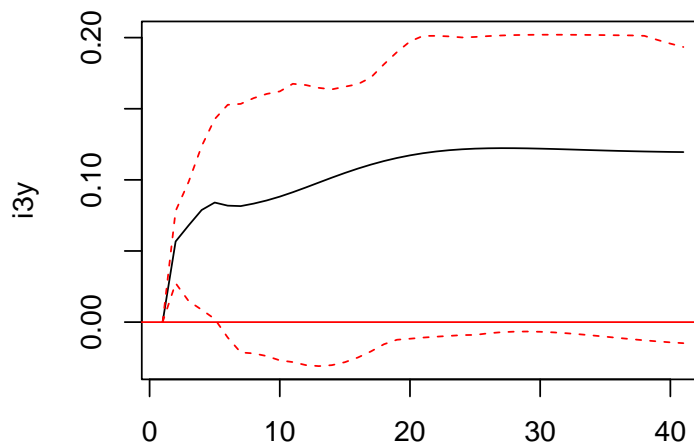
95 % Bootstrap CI, 100 runs

Assignment Project Exam Help

VECM(3, 2): Response of i3y to a shock in i5y

<https://tutorcs.com>

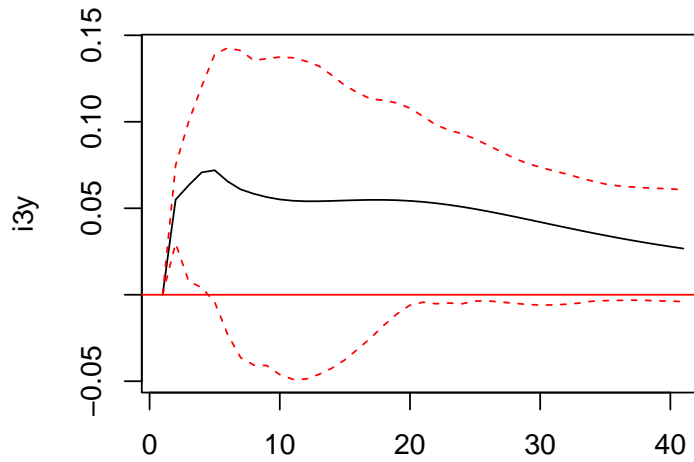
VECM(3, 3): Response of i3y to a shock in i5y



95 % Bootstrap CI, 100 runs

VECM(3, 3): Response of i3y to a shock in i5y

VECM(3, 4): Response of i3y to a shock in i5y



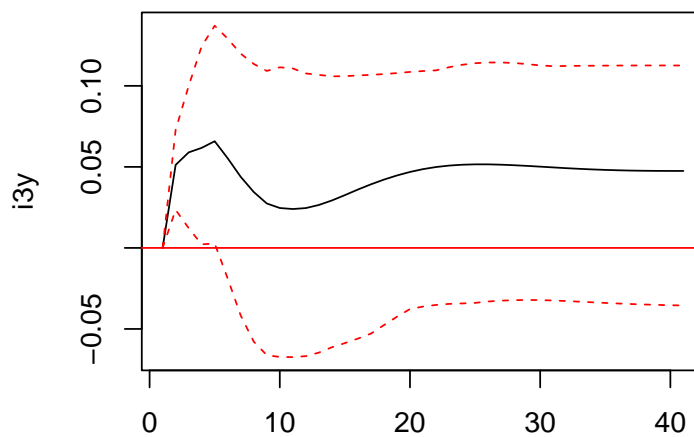
Assignment Project Exam Help

VECM(3, 4): Response of i3y to a shock in i5y

<https://tutorcs.com>

WeChat: cstutorcs

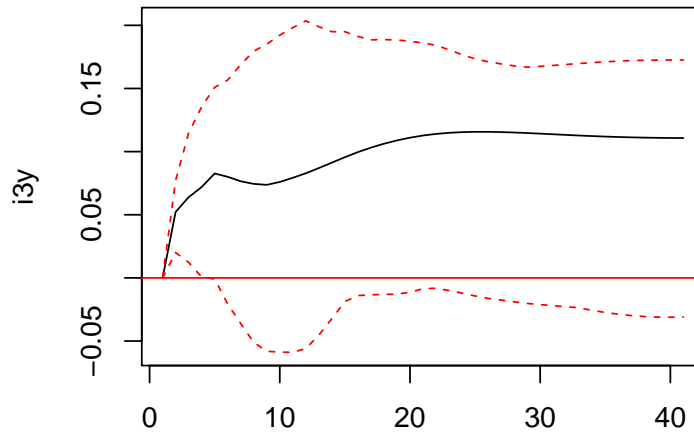
VECM(4, 2): Response of i3y to a shock in i5y



95 % Bootstrap CI, 100 runs

VECM(4, 2): Response of i3y to a shock in i5y

VECM(4, 3): Response of i3y to a shock in i5y



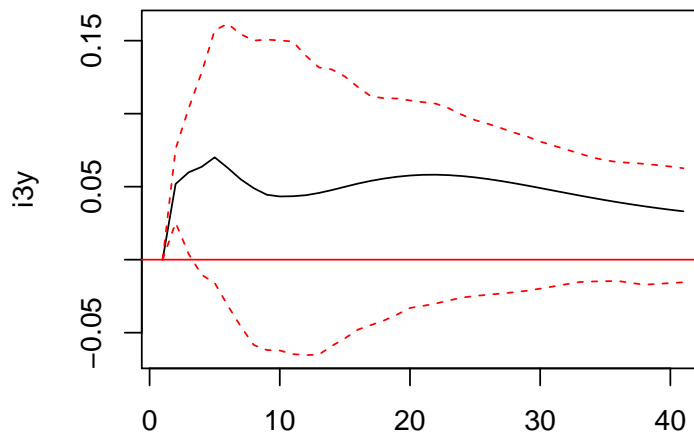
Assignment Project Exam Help

VECM(4, 3): Response of i3y to a shock in i5y

<https://tutorcs.com>

WeChat: cstutorcs

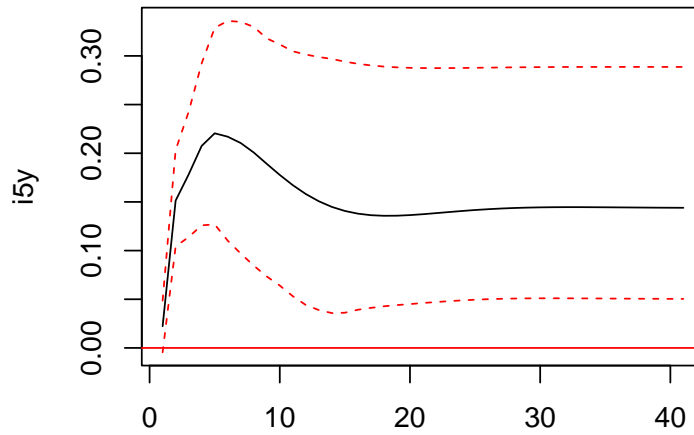
VECM(4, 4): Response of i3y to a shock in i5y



95 % Bootstrap CI, 100 runs

VECM(4, 4): Response of i3y to a shock in i5y

VECM(3, 2): Response of i5y to a shock in i90d



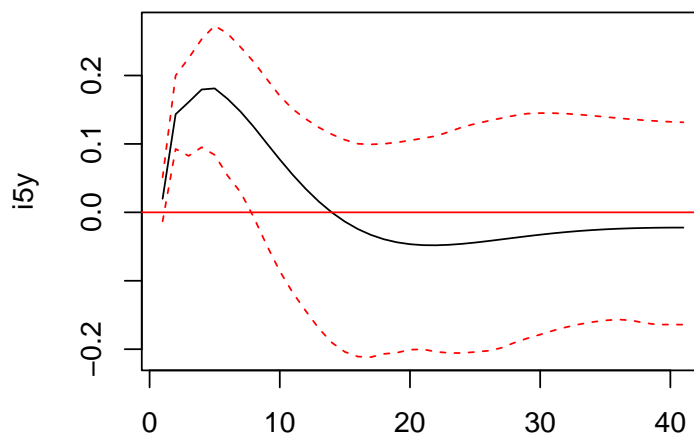
Assignment Project Exam Help

95 % Bootstrap CI, 100 runs

VECM(3, 2): Response of i5y to a shock in i90d

<https://tutorcs.com>

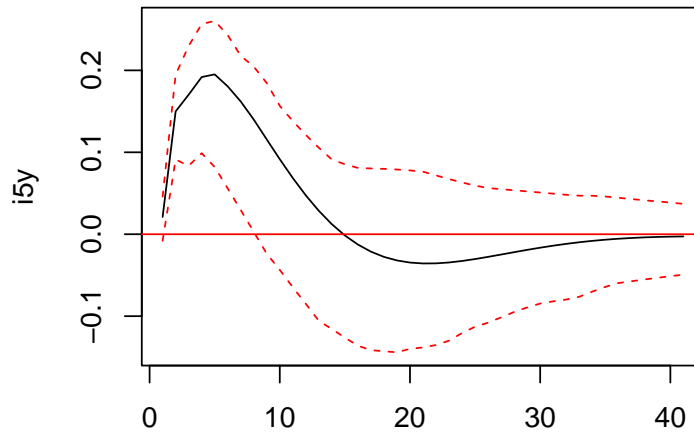
VECM(3, 3): Response of i5y to a shock in i90d



95 % Bootstrap CI, 100 runs

VECM(3, 3): Response of i5y to a shock in i90d

VECM(3, 4): Response of i5y to a shock in i90d



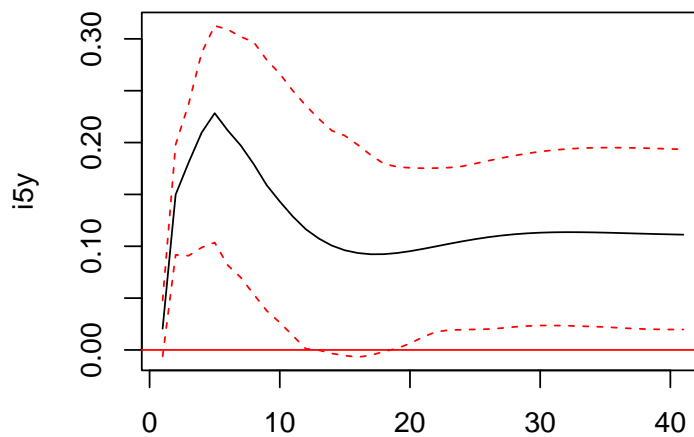
Assignment Project Exam Help

VECM(3, 4): Response of i5y to a shock in i90d

<https://tutorcs.com>

WeChat: cstutorcs

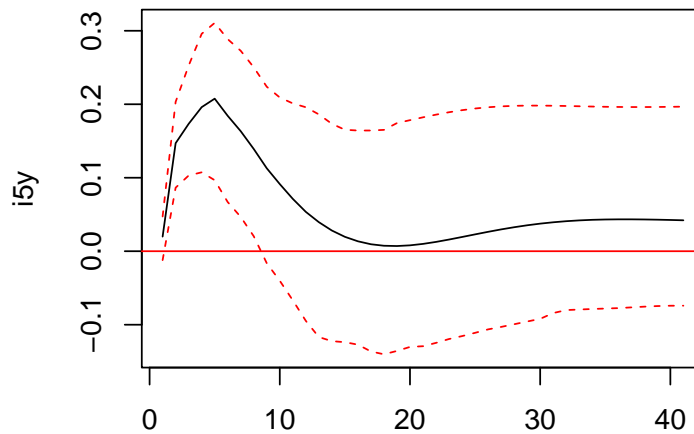
VECM(4, 2): Response of i5y to a shock in i90d



95 % Bootstrap CI, 100 runs

VECM(4, 2): Response of i5y to a shock in i90d

VECM(4, 3): Response of i5y to a shock in i90d



95 % Bootstrap CI, 100 runs

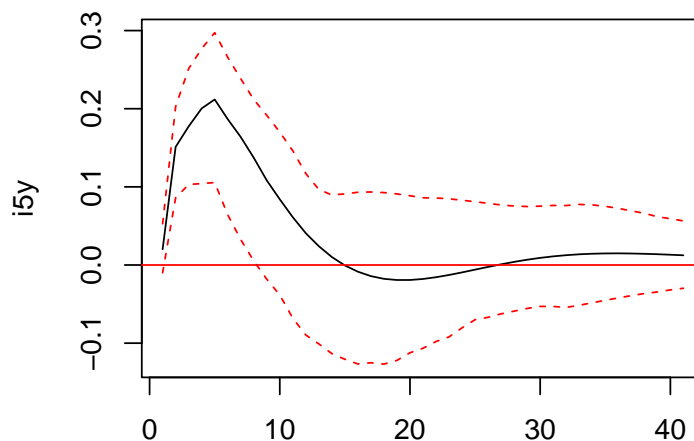
Assignment Project Exam Help

VECM(4, 3): Response of i5y to a shock in i90d

<https://tutorcs.com>

WeChat: cstutorcs

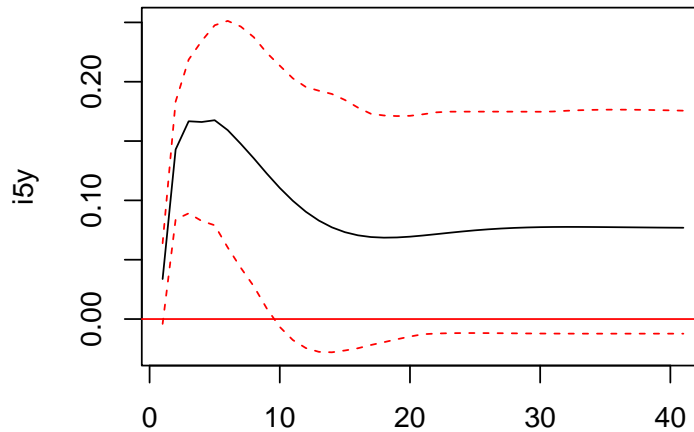
VECM(4, 4): Response of i5y to a shock in i90d



95 % Bootstrap CI, 100 runs

VECM(4, 4): Response of i5y to a shock in i90d

VECM(3, 2): Response of i5y to a shock in i180d



95 % Bootstrap CI, 100 runs

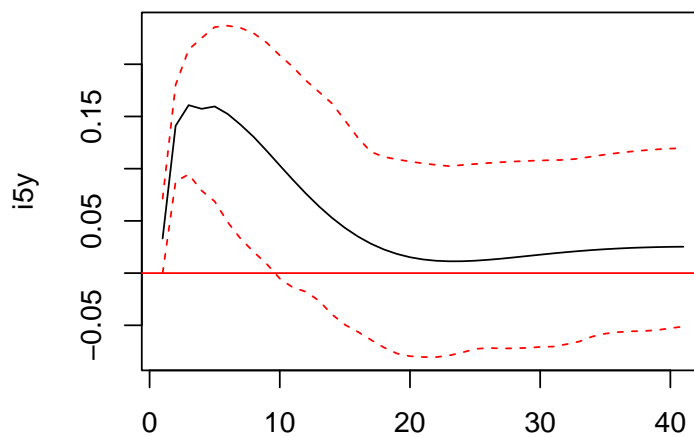
Assignment Project Exam Help

VECM(3, 2): Response of i5y to a shock in i180d

<https://tutorcs.com>

VECM(3, 3): Response of i5y to a shock in i180d

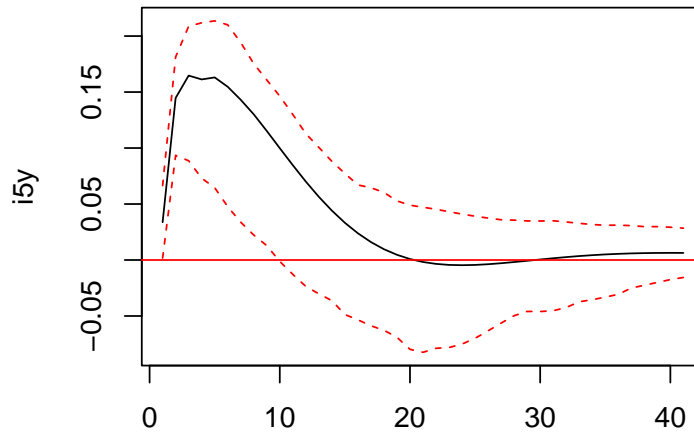
WeChat: cstutorcs



95 % Bootstrap CI, 100 runs

VECM(3, 3): Response of i5y to a shock in i180d

VECM(3, 4): Response of i5y to a shock in i180d



95 % Bootstrap CI, 100 runs

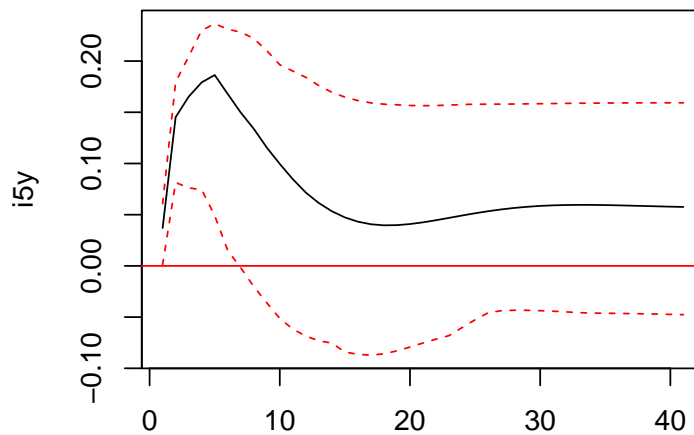
Assignment Project Exam Help

VECM(3, 4): Response of i5y to a shock in i180d

<https://tutorcs.com>

VECM(4, 2): Response of i5y to a shock in i180d

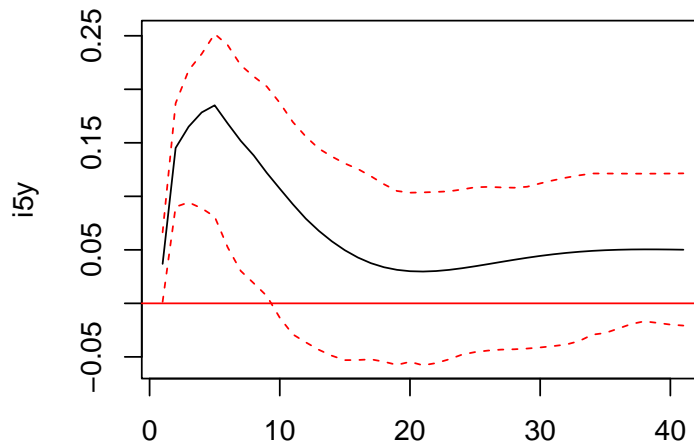
WeChat: cstutorcs



95 % Bootstrap CI, 100 runs

VECM(4, 2): Response of i5y to a shock in i180d

VECM(4, 3): Response of i5y to a shock in i180d

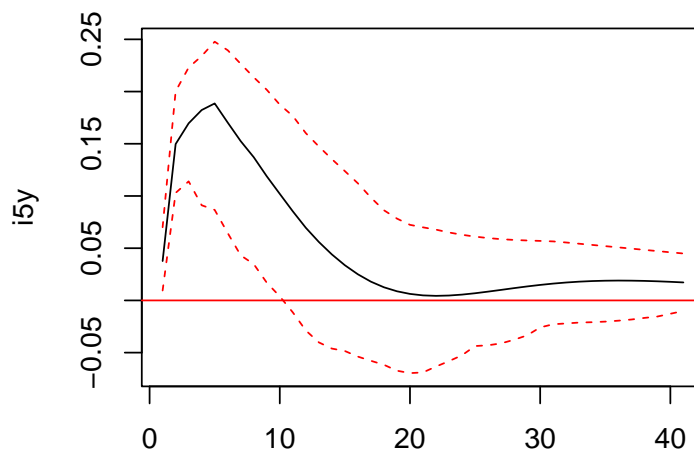


Assignment Project Exam Help

VECM(4, 3): Response of i5y to a shock in i180d

<https://tutorcs.com>

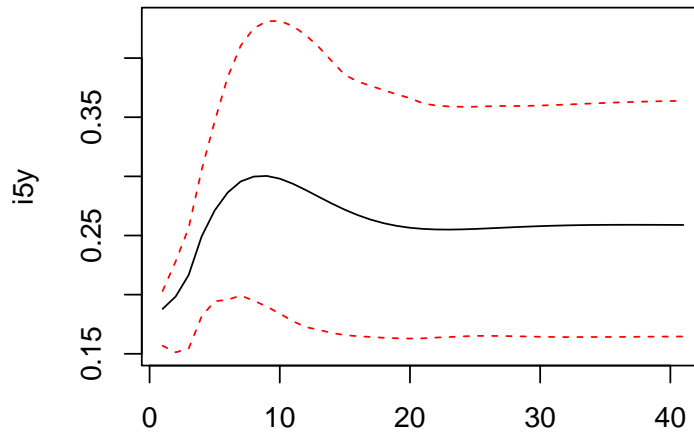
WeChat: cstutorcs



95 % Bootstrap CI, 100 runs

VECM(4, 4): Response of i5y to a shock in i180d

VECM(3, 2): Response of i5y to a shock in i3y



Assignment Project Exam Help

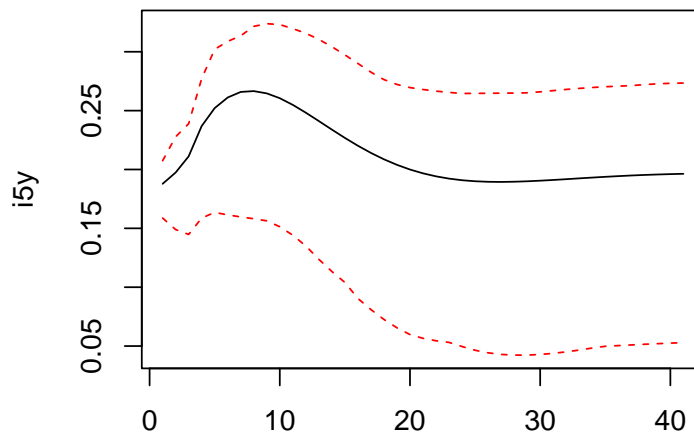
95 % Bootstrap CI, 100 runs

VECM(3, 2): Response of i5y to a shock in i3y

<https://tutorcs.com>

WeChat: cstutorcs

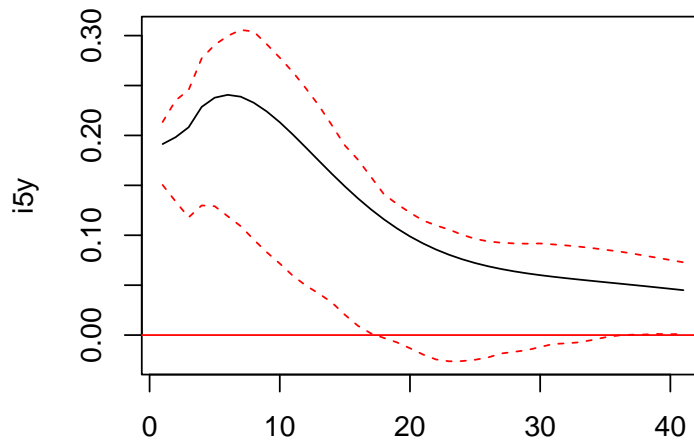
VECM(3, 3): Response of i5y to a shock in i3y



95 % Bootstrap CI, 100 runs

VECM(3, 3): Response of i5y to a shock in i3y

VECM(3, 4): Response of i5y to a shock in i3y



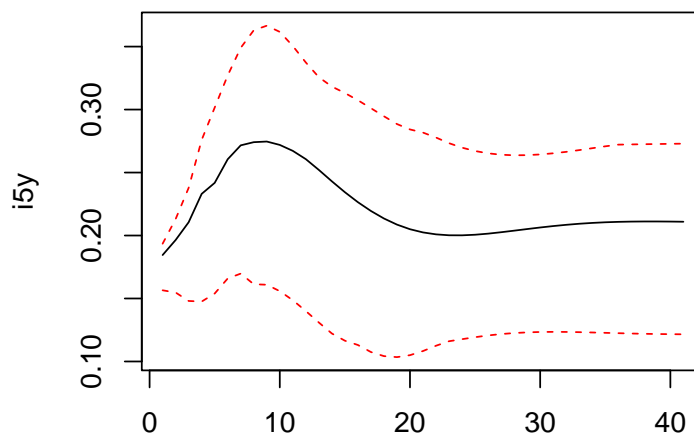
Assignment Project Exam Help

VECM(3, 4): Response of i5y to a shock in i3y

<https://tutorcs.com>

WeChat: cstutorcs

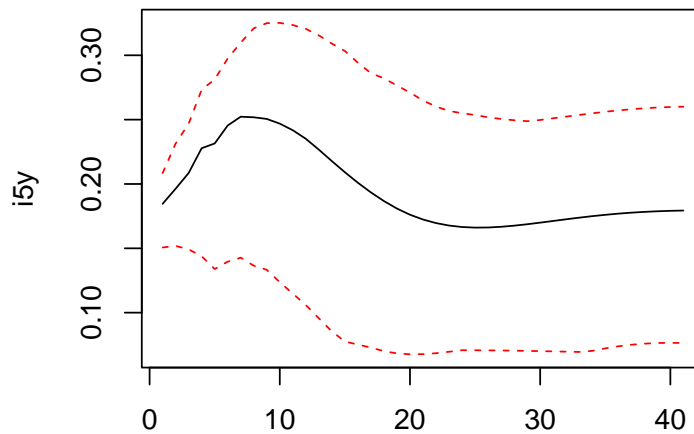
VECM(4, 2): Response of i5y to a shock in i3y



95 % Bootstrap CI, 100 runs

VECM(4, 2): Response of i5y to a shock in i3y

VECM(4, 3): Response of i5y to a shock in i3y



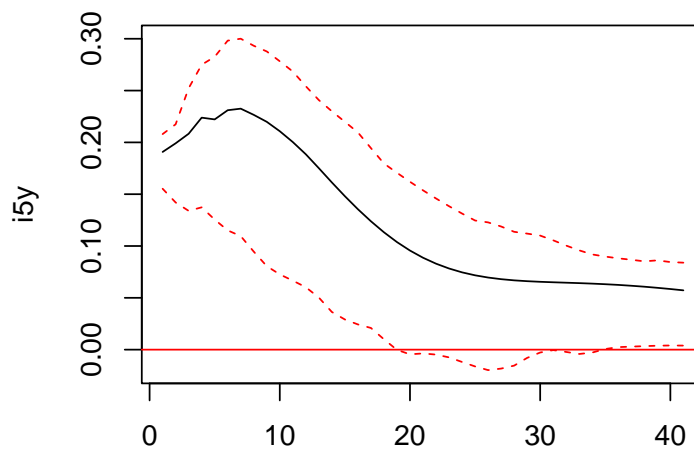
Assignment Project Exam Help

VECM(4, 3): Response of i5y to a shock in i3y

<https://tutorcs.com>

WeChat: cstutorcs

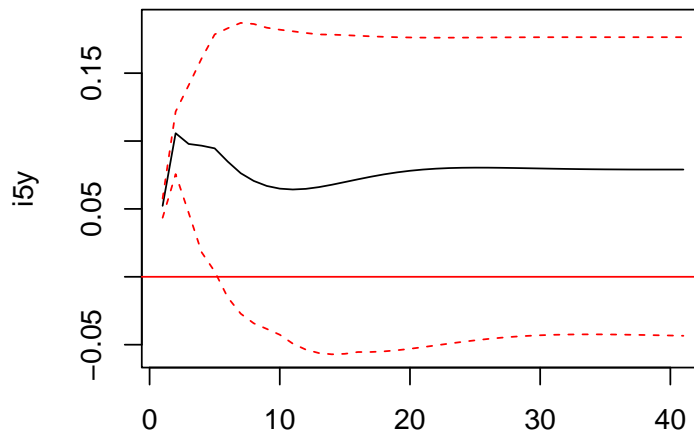
VECM(4, 4): Response of i5y to a shock in i3y



95 % Bootstrap CI, 100 runs

VECM(4, 4): Response of i5y to a shock in i3y

VECM(3, 2): Response of i5y to a shock in i5y



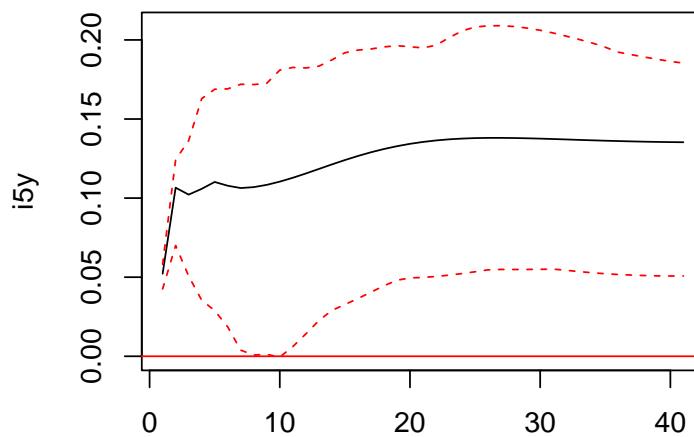
Assignment Project Exam Help

VECM(3, 2): Response of i5y to a shock in i5y

<https://tutorcs.com>

WeChat: cstutorcs

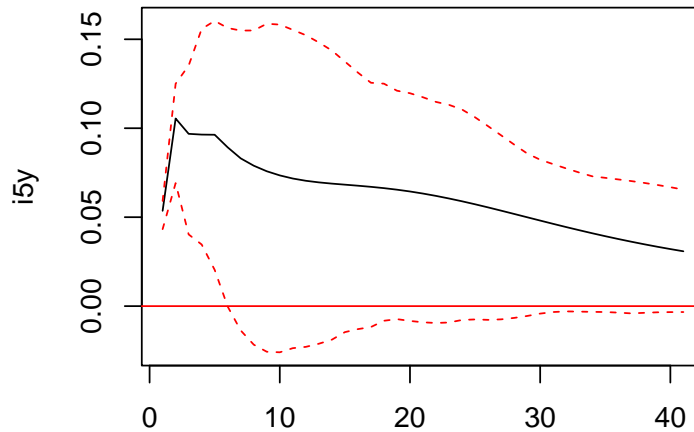
VECM(3, 3): Response of i5y to a shock in i5y



95 % Bootstrap CI, 100 runs

VECM(3, 3): Response of i5y to a shock in i5y

VECM(3, 4): Response of i5y to a shock in i5y



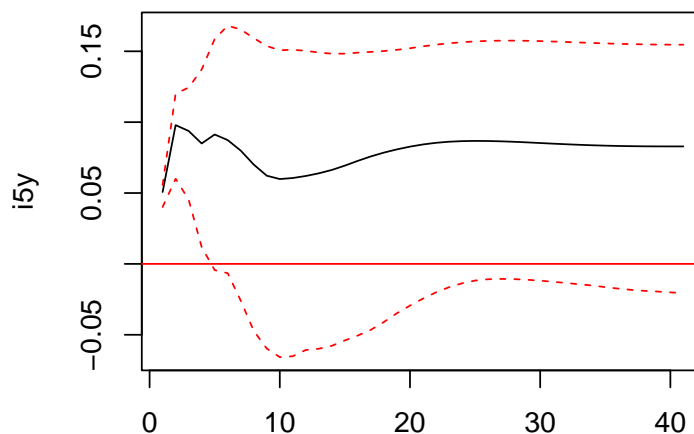
Assignment Project Exam Help

VECM(3, 4): Response of i5y to a shock in i5y

<https://tutorcs.com>

WeChat: cstutorcs

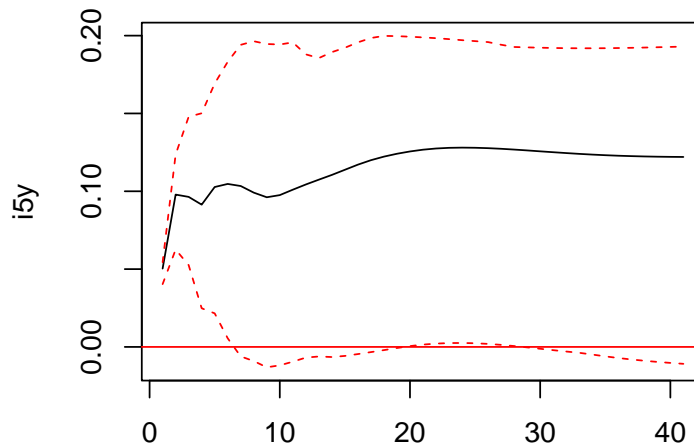
VECM(4, 2): Response of i5y to a shock in i5y



95 % Bootstrap CI, 100 runs

VECM(4, 2): Response of i5y to a shock in i5y

VECM(4, 3): Response of i5y to a shock in i5y



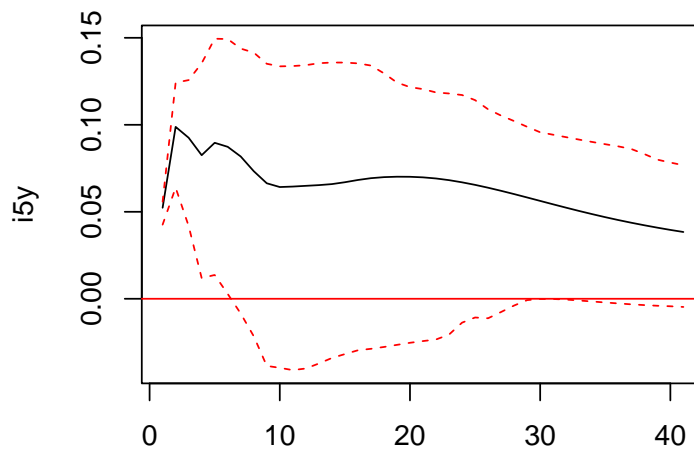
Assignment Project Exam Help

VECM(4, 3): Response of i5y to a shock in i5y

<https://tutorcs.com>

WeChat: cstutorcs

VECM(4, 4): Response of i5y to a shock in i5y



95 % Bootstrap CI, 100 runs

VECM(4, 4): Response of i5y to a shock in i5y

What really stands out from these plots is how sensitive inference from IRFs is to

specifications in terms of r . This underscores the pitfall of blindly relying on a naive approach and ignoring specification uncertainty!

For example, consider the responses of $i5y_t$. Looking first at the responses to a $i90d_t$ specific shocks, we find that short run responses (up to around 8 quarters) are generally similar across different specifications. However, long-run responses are drastically different for $r = 2$ models compared to $r = 3$ and $r = 4$. A similar story emerges in the responses of $i5y_t$ to $i180d_t$ as well.

Now, if we consider the naive approach of choosing a particular r based on the rule that it is the lowest rank for which H_0 cannot be rejected by Johansen's trace test, we would select $r = 2$ and ignore completely $r = 3$ and $r = 4$ as possibly valid specifications. The IRFs presented here make it clear how we may be led quite astray in our inference with the naive approach. Ignoring the responses with $r = 3$ and $r = 4$ would indeed be quite misleading.

Interestingly, the responses of $i5y_t$ to the $i3y_t$ specific shock are more similar between the $r = 2$ and $r = 3$ specifications, but differ in the long-run response obtained from the unrestricted ($r = 4$) specification. Again, selecting any one r and ignoring uncertainty associated with alternative rank specifications can be very detrimental.

Being transparent and interpreting results in light of this uncertainty is far more informative. In this case, we can generally conclude that inference on short-run responses entails a higher degree of confidence, but long-run responses are subject to substantial specification uncertainty.

Assignment Project Exam Help

<https://tutorcs.com>

WeChat: cstutorcs