

The Macroeconomic Effects of Fiscal Adjustments in The UK

Samid Ali

Contents

1	Abstract	3
2	Introduction	3
3	Literature Review	3
3.1	Costs of high indebtedness	3
3.2	Fiscal Consolidaton	4
3.3	Synthesis of Methodology	4
3.4	Data Proc above	5
4	Econometric Methodology	5
4.1	Model	6
5	Identification	8
6	Results	10
6.1	Unrestricted Model	10
6.2	IRFs	10
6.3	Chow test for Structural Breaks	15
7	Robustness	18
7.1	Stability	18
7.2	Normality	18
7.3	Lag length	18
7.4	Identification	18

8 Discussion/ Policy Implications	18
8.1 Limitations	18
9 Conclusion	19
10 Bibliography	19
11 Data Appendix	19
11.1 Data Sources	19
11.2 Data Plots	20
12 Technical Appendix	23
12.1 VECM Code/ Results	23
12.2 IRFs	28
12.3 FEVDs post GFC:	30

1 Abstract

2 Introduction

3 Literature Review

3.1 Costs of high indebtedness

Sutherland, Hoeller, and Merola [3] draw attention to the fiscal challenges facing countries following the Global Financial Crisis, noting that gross government debt has exceeded 100% of GDP for the OECD as an aggregate. This has been exacerbated following the Covid pandemic where governments implemented fiscal measures to mitigate the economic costs of the pandemic. Makin and Layton (2021) highlight that governments must employ fiscal responsibility to protect their economies from the risks that high indebtedness exposes them to. Warmedinger, Checherita-Westphal, and De Cos [4] emphasise the importance of public debt sustainability for ensuring macroeconomic stability. There are several mechanisms through which excessive debt could harm the economy. For instance, concerns regarding public finances could reduce business confidence, leading to decreased investment and thus a slowdown in growth. Additionally, strained public finances could hinder the ability of economies to react counter cyclically to economic shocks. This rationale is supported by the IMF (2023) who argue that economies should rebuild their fiscal buffers to reduce their debt vulnerabilities. Therefore, fiscal consolidation is clearly needed to ensure the long-term resilience of the economy. As a target, Sutherland, Hoeller, and Merola [3] propose that countries should aim to bring debt levels towards 50% of GDP: a figure which would require the UK to halve its current debt levels (ONS, 2025). The IMF (2023) argues that to stabilise GDP, fiscal adjustments should be in the region of up to 4% of GDP. Thus, achieving this objective would require significant fiscal adjustments, motivating further research to support the transition towards more sustainable public finances. Alesina, Favero, and Giavazzi (2012) find that permanent fiscal adjustments have lower output costs, interpreting this result as due to business confidence. (easier to forecast when fiscal adjustments are more predictable? Thus have less of a effect on confidence)

Kumar and Woo (2015) find that greater indebtedness is associated with lower economic growth. They also find noticeable nonlinearities in this result, with the most severe effect when public indebtedness exceeds 90% of GDP. Given the aforementioned level of public indebtedness in the UK ...

Blanchard (2019) argue that even when the interest rate is less than the growth rate, and thus there is no fiscal cost of high indebtedness, there may still be a welfare cost due to reduced capital accumulation.

3.2 Fiscal Consolidation

While the importance of fiscal consolidation has been highlighted, it is crucial that these measures are not at the expense of the broader economy. By investigating forecast errors for a sample of European countries, Blanchard and Leigh [1] find that larger anticipated fiscal consolidation was associated with lower growth. This result was interpreted as due to the fiscal multiplier being greater than anticipated by forecasters. Consequently, fiscal tightening would have further dampened demand, and thus improvements in fiscal consolidation would be offset by reduced growth. Gechert (2019) adopt a similar methodology, finding that austerity measures in the Euro Area deepened the crisis, contributing to hysteresis effects. Fatas and Summers (2018) extend this research, investigating the long-term effects that fiscal adjustments have had on GDP. Their analysis suggests that fiscal consolidations have failed to lower the debt-to-GDP ratio due to a hysteresis effect of contractionary fiscal policy. This research underscores the need for effectively quantifying fiscal multipliers to understand potential trade-offs between various economic objectives. Ilzetzki, Mendoza, and Végh [2] provide further insight into the fiscal multiplier, suggesting that the heterogeneity in the estimates reported in the literature can be attributed to differences in structural characteristics of the economy considered. This reinforces the importance of research to better understand the fiscal multiplier for different policy instruments, particularly as this may vary across countries and over time. Additionally, Alesina et al (2015) compare multipliers due to spending and tax adjustments. They find that ... have more severe effects, attributing this to reduced business confidence. Alesina et al (2002) investigate the effect of fiscal policy on investment.

3.3 Synthesis of Methodology

Capek and Cuasera (2020) simulate 20 million fiscal multipliers, highlighting how methodological choices contribute to the heterogeneity in estimates of fiscal multipliers prevalent in the literature. Consequently, they advocate for explicitly outlining modelling choices and assumptions. Similarly, Gechert (2017) provides a synthesis of the methodologies used to estimate fiscal multipliers, highlighting competing definitions for the fiscal multiplier and possible issues in its estimation. Among these issues, Gechert (2017) highlight potential omitted variable bias in the VAR model (motivating the use of additional controls such as the price level and real interest rate), anticipation effects (cf Leeper + Zha fiscal foresight), and nonlinearities.

Structural Vector Autoregressions (SVARs) have been prominent in the literature to estimate fiscal multipliers. Various approaches to identification have been used, with XXX (YYYY) noting that after accounting for the empirical specification, the competing identifying approaches have little effect on the estimated multipliers. Blanchard and Perotti (2002) pioneered this strand of

the research, leveraging methodologies previously popularised by the monetary economics. To identify their SVAR, Blanchard and Perotti leverage institutional information. They provide a definition for the fiscal variables and highlight that government expenditure is predetermined within a quarter. Recursive measures to identification have been employed by Fatas and Mihov (YYYY) and Fernandez (2008). Fernandez argues that Uhlig and Mountford (200Y) apply restrictions on the signs of the impulse response functions. Caldara and Kamps (2008) reviews the literature on SVAR identification. Caldara and Kamps (2017) introduce a new approach for identification.

Add DSGE lit for context

3.4 Data Proc above

4 Econometric Methodology

In this section we outline the empirical methodology, as well as the data employed by this research. We use quarterly data from 1987:1 to 2023:3, modelling this using a six-dimensional VAR at with a lag length of 4. This lag order follows Blanchard and Perroti (2002), and suggests that past values of the endogenous variables continue to have an effect for up to a year. While statistical procedures were not used to determine this lag order, it has an econometric rationale: Kilian and Lutkepohl (2016) note that due to poor finite sample properties of lag selection techniques, it may be appropriate to impose a fixed lag length. Particularly when concerned with impulse response analysis, the risks of underestimating the lag length exceed those of using a larger order which may better reflect the dynamics of the data at the cost of inefficient estimates. Given these considerations, and the degrees of freedom restrictions imposed by the sample size, the lag order of 4 was determined.

Regarding the endogenous variables considered, Blanchard and Perotti (2002) investigate the effects of fiscal shocks using a three-dimensional VAR model consisting of GDP, government expenditure, and government revenue. While such a model could be used to estimate the effects of fiscal shocks, Gechert (2017) highlights the potential issues of omitted variable bias. Therefore, we augment the model to include also a short term interest rate, the GDP deflator rate, and UK exports. These variables are included to account for the effects of monetary policy, price levels, and trade respectively. Consequently, the impulse response functions reported later are better interpreted as the response of GDP to the fiscal variables *ceteris paribus*.

Consistent with Fernández (2006), all variables are log-transformed prior to estimation, except for the short-term interest rate, which enters the model in levels. Furthermore, the fiscal variables and GDP are used in real terms. Capek and Cuaresma (2020) highlight that data used to estimate fiscal multipliers is typically seasonally adjusted. Therefore (with the exception of GDP which was

sourced after seasonal adjustment) variables have been seasonally adjusted using the X-13ARIMA-SEATS method. This reduces the noise in the data, allowing for more meaningful results. The fiscal variables are defined at the general level and follow the European System of Accounts (ESA, 2010). In particular, government expenditure represents the outflows associated with government activities, including consumption, investment, and transfers. The inflows to the government, government revenue, consists of receipts net of transfer and interest payments. *Cf Blanchard and Perotti (2002)*.

- debt sustainability (inflation/ interest rate changing real cost of debt despite no change in deficit ceteris paribus)

4.1 Model

Note: VAR analysis requires stability of the system.

4.1.1 VECM

The reduced-form VAR model can be written as:

$$X_t = \mu_0 + \mu_1 t + A_1 X_{t-1} + A_2 X_{t-2} + \dots + X_p Y_{t-p} + \epsilon_t \quad (1)$$

where we have:

$$X_t = \begin{pmatrix} G_t \\ R_t \\ GDP_t \\ \tau_t \\ P_t \end{pmatrix}$$

The terms $\mu_0 + \mu_1 t$ in equation 1 capture the deterministic component of the model, these will be further refined after we highlight the features of the data.

The VAR model above assumes that the data generating process includes a deterministic linear time trend and constant intercept. This is included to mitigate for potential spurious regression between trending factors in the endogenous variables and to account for their nonzero mean.

Killian and Lutkepohl (2016) show that the VAR model can be reparameterised as a Vector Error Correction Model (VECM) by subtracting the lagged variables and rearranging terms. When correctly specified, this representation can yield efficiency gains: including the error correction term may allow the model to capture important dynamics in the long run relationships between variables. Therefore we proceed with tests of cointegration to determine the empirical specification employed by this research.

$$\Delta X_t = \Pi X_{t-1} + \sum_{i=1}^{k-1} \Gamma_i \Delta X_{t-i} + \mu + \varepsilon_t \quad (2)$$

Hubrich, Lutkepohl, and Saikkonen (2001) provide a review of cointegration tests. Conducting a monte carlo simulation, they find that the LR type tests are among the best performing in practice, with other classes of tests having poor power or size. Gonzalo and Lee (1998) potential pitfalls of the Johanesen LR test for cointegration

Hubrich, Lutkepohl, and Saikkonen (1998) highlight that improperly specified deterministic terms invalidate cointegration testing, with reduced power.

In the cointegration test we assume a constant deterministic term. The literature highlights the importance of ensuring any deterministic terms are properly specified for the validity of cointegration tests.

We perform Johansen’s maximum eigenvalue test for cointegration, which compares the null hypothesis of rank r , to the alternative of rank $r+1$. As can be seen from the table, there is insufficient evidence to reject the null hypothesis of no cointegrating relations at the 5% levels. Consequently, the VECM reduces to a VAR in the differences of variables (Kilian and Lutkepohl, 2016). For robustness, we also perform Johansen’s trace test, which has as the alternative hypothesis that the rank is greater than assumed under the null hypothesis. Under this test we reject at the 5% level that there are no cointegrating relations, and at the same level fail to reject that there is a single cointegrating relation.

Cheung and Lai (1993), using a Monte Carlo simulation, find that there is finite sample bias in Johansen’s LR test, with the tests being biased towards finding cointegration at a rate greater than implied by asymptotic theory. Additionally, Lutkepohl, Saikkonen, and Trenkler (2001) find that in small samples, while trace tests had size distortions, they had power advantages relative to the maximum eigenvalue test. This suggests that there may exist a statistical basis to support the hypothesis of no cointegrating relationship.

Furthermore, Kilian and Lutkepohl (2016) highlight the asymmetric consequences of imposing a unit root. When the underlying data generating process possesses a unit root, the reduced form model can benefit from increased efficiency in estimation by imposing a unit root. Failing to impose a unit root in such a case would only reduce the precision of the Least Squares estimates. In contrast, when the underlying process does not follow a unit root, incorrectly imposing one would result in over-differencing - resulting in an inconsistent estimator. Given the parsimony of a VAR in differences and theoretical support for the maximal eigenvalue test we proceed in this analysis by assuming there are no cointegrating relationships.

The lag order for the VAR in differences is $p-1$, therefore we proceed with a lag order of 3

One such relationship that could be argued a priori is a cointegrating relationship between government revenue and spending. Intuitively, this would mean that while there can be shocks will lead the variables back to their long run equilibrium relationship. This could be interpreted as stationarity of the government deficit

Add code results to appendix?

NB on unit root tests: failing to reject H_0 does not mean we accept H_0 !!!!

Cheung and Lai (1993) perform a Monte Carlo analysis to evaluate the finite sample properties of Johansen's LR tests. They find that while the trace and maximum eigenvalue tests perform similarly, they are biased

For robustness we also consider the VAR in levels.

For robustness we also perform Phillips-Ouliaris' (1990) residual based test for cointegration. Using this test we again fail to reject the null hypothesis of no cointegration, suggesting that a VECM is not an appropriate representation for our data.

5 Identification

Explaining distinction between the reduced form and structural model. Interested in the structural shocks which we will recover as follows.

Let X_t be the vector of variables:

$$X_t = \begin{pmatrix} G_t \\ R_t \\ GDP_t \\ T_t \\ P_t \end{pmatrix}$$

The reduced-form VAR model can be written as:

$$X_t = A_1 X_{t-1} + A_2 X_{t-2} + \dots + A_p X_{t-p} + \epsilon_t$$

where ϵ_t is the vector of reduced-form residuals. To recover the structural shocks u_t , we assume:

$$\epsilon_t = B u_t$$

B^{-1} is the structural impact multiplier matrix.

where B is a lower triangular matrix. The structural shocks u_t are assumed to be uncorrelated and have unit variance.

The matrix B can be obtained using Cholesky decomposition of the covariance matrix of the reduced-form residuals Σ_ϵ :

$$\Sigma_\epsilon = E[\epsilon_t \epsilon_t'] = BB'$$

Given the recursive ordering (G, R, GDP, T, P) , the matrix B has the form:

$$B = \begin{pmatrix} b_{11} & 0 & 0 & 0 & 0 \\ b_{21} & b_{22} & 0 & 0 & 0 \\ b_{31} & b_{32} & b_{33} & 0 & 0 \\ b_{41} & b_{42} & b_{43} & b_{44} & 0 \\ b_{51} & b_{52} & b_{53} & b_{54} & b_{55} \end{pmatrix}$$

Thus, the structural shocks u_t can be recovered as:

$$u_t = B^{-1}\epsilon_t$$

Killian and Lutkepohl (2017) highlight that identification of the structural parameters is not a purely statistical concern, the restrictions must also be economically meaningful for the resulting structural parameters to be identified. Therefore we proceed with an exposition of the economic assumptions implicit in the impact multiplier matrix, B , comparable to Fernandez (2006).

- 1) Blanchard and Perotti (2002) argue that the use of quarterly data allows government spending to be interpreted as predetermined with respect to the rest of the variables within the quarter. This is motivated by implementation lags for changes to government spending and consequently this is ordered first.
- 2) Given physical constraints, the interest rate is assumed not to react contemporaneously to price, net taxes, output, or exports. Thus the short term rate is considered the next most exogenous variable.
- 3) However monetary policy shocks are assumed to affect output, net taxes, and prices contemporaneously. Fernandez (2006) justifies this assumption by noting that interest movements are anticipated and thus they can be transmitted to real variables relatively quickly.
NB on the appropriateness of the assumption that interest rate does not react to price/ output.

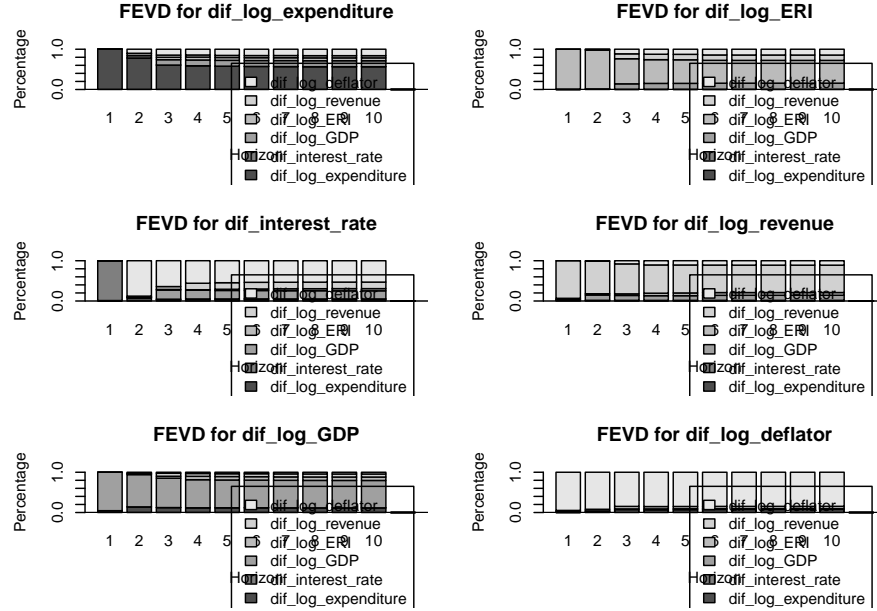
3a) By construction exports contemporaneously affect GDP and revenue. Additionally exports affect the price level through

- 4) Due to price stickiness, prices do not react contemporaneously to shocks to GDP,
- 5) Due to physical constraints in adjusting consumption and investment, net taxes are assumed not to affect economic activity.

6 Results

6.1 Unrestricted Model

6.1.1 FEVDs



Discuss FEVDs: We report the FEVDs, while 95% of the variability in the difference in log GDP a quarter after a shock is explained by itself, by 10 quarters, this percentage reduces to 68%. At 19%, the difference in the logarithms of the fiscal variables explain a large amount of the remaining variability.

6.2 IRFs

A standard deviation shock to the difference in log expenditure results in a -0.22 standard deviation decrease in the difference in logarithms of GDP. Notably, the sign of the IRF at index 0 is negative, suggesting that an acceleration in expenditure has a crowding out effect.

Table 1: Cumulative IRF by Horizon and Variable

Horizon	Variable	IRF
0	dif_log_expenditure	-0.22
1	dif_log_expenditure	0.14

Horizon	Variable	IRF
2	dif_log_expenditure	0.08
3	dif_log_expenditure	0.04
4	dif_log_expenditure	0.10
5	dif_log_expenditure	0.12
6	dif_log_expenditure	0.11
7	dif_log_expenditure	0.12
8	dif_log_expenditure	0.12
9	dif_log_expenditure	0.13
10	dif_log_expenditure	0.13
0	dif_log_revenue	0.00
1	dif_log_revenue	-0.14
2	dif_log_revenue	0.16
3	dif_log_revenue	0.17
4	dif_log_revenue	0.17
5	dif_log_revenue	0.09
6	dif_log_revenue	0.16
7	dif_log_revenue	0.15
8	dif_log_revenue	0.12
9	dif_log_revenue	0.12
10	dif_log_revenue	0.14

We report the cumulated structural impulse response functions up to a horizon of 10 quarters and include bands for the 68% confidence interval, approximately corresponding to a one standard deviation interval. As nonlinear functions of the parameters, The confidence intervals are obtained

```
## $dif_log_expenditure
##      dif_log_expenditure dif_interest_rate dif_log_GDP dif_log_ERI
## [1,]      1.0000000      0.000000e+00   0.0000000   0.0000000
## [2,]      0.7828833      1.117217e-05   0.0565156   0.05701155
## [3,]      0.6070228      1.373020e-05   0.1330038   0.05977481
## [4,]      0.5895910      1.381450e-05   0.1429120   0.07196719
## [5,]      0.5790353      1.393819e-05   0.1444186   0.07100721
## [6,]      0.5729853      1.411122e-05   0.1457577   0.07102662
## [7,]      0.5692544      1.530939e-05   0.1485466   0.07093461
## [8,]      0.5666189      1.525315e-05   0.1478854   0.07106259
## [9,]      0.5664138      1.526633e-05   0.1480003   0.07106955
## [10,]     0.5661918      1.533691e-05   0.1480792   0.07113198
##      dif_log_revenue dif_log_deflator
## [1,]      0.000000000      0.0000000
## [2,]      0.002588732      0.1009897
## [3,]      0.054081686      0.1461031
## [4,]      0.052603365      0.1429127
```

```

## [5,]      0.052168615      0.1533563
## [6,]      0.053179162      0.1570371
## [7,]      0.054527117      0.1567219
## [8,]      0.054301939      0.1601159
## [9,]      0.054291091      0.1602099
## [10,]     0.054270504      0.1603112
##
## $dif_interest_rate
##      dif_log_expenditure dif_interest_rate dif_log_GDP dif_log_ERI
## [1,]      0.01177570      0.988224298  0.00000000  0.00000000
## [2,]      0.07759661      0.009039674  0.00434749  0.01842520
## [3,]      0.04042069      0.004848772  0.22907875  0.01379291
## [4,]      0.04674809      0.004172448  0.21632783  0.01301441
## [5,]      0.04722316      0.004070783  0.21039259  0.03046427
## [6,]      0.04847900      0.003998084  0.20754361  0.03851606
## [7,]      0.04810147      0.003911849  0.20316441  0.04822280
## [8,]      0.04776211      0.003877103  0.20304267  0.05071482
## [9,]      0.04752042      0.003856673  0.20197451  0.05398684
## [10,]     0.04751458      0.003854756  0.20182320  0.05409419
##      dif_log_revenue dif_log_deflator
## [1,]      0.00000000      0.00000000
## [2,]      0.01160525      0.8789858
## [3,]      0.07062771      0.6412312
## [4,]      0.16071506      0.5590222
## [5,]      0.16229921      0.5455500
## [6,]      0.16553207      0.5359312
## [7,]      0.16219903      0.5344004
## [8,]      0.16256526      0.5320380
## [9,]      0.16216652      0.5304950
## [10,]     0.16203027      0.5306830
##
## $dif_log_GDP
##      dif_log_expenditure dif_interest_rate dif_log_GDP dif_log_ERI
## [1,]      0.04553131      2.927939e-05  0.9544394  0.00000000
## [2,]      0.14057000      4.098889e-05  0.8007930  0.02022823
## [3,]      0.12293894      5.016312e-05  0.7297122  0.04769893
## [4,]      0.11746427      4.831355e-05  0.6968085  0.08430538
## [5,]      0.11806377      4.800228e-05  0.6895120  0.08587197
## [6,]      0.11775972      5.120631e-05  0.6865309  0.08557455
## [7,]      0.11656332      5.285824e-05  0.6829253  0.08484441
## [8,]      0.11626149      5.277415e-05  0.6809989  0.08525223
## [9,]      0.11611036      5.288549e-05  0.6802377  0.08527051
## [10,]     0.11605571      5.299709e-05  0.6799019  0.08527095
##      dif_log_revenue dif_log_deflator
## [1,]      0.00000000      0.00000000
## [2,]      0.01456208      0.02380570

```

```

## [3,]      0.07370665      0.02589309
## [4,]      0.06996599      0.03140751
## [5,]      0.06914238      0.03736192
## [6,]      0.07288960      0.03719402
## [7,]      0.07510101      0.04051314
## [8,]      0.07501037      0.04242423
## [9,]      0.07530829      0.04302022
## [10,]     0.07525295      0.04346548
##
## $dif_log_ERI
##      dif_log_expenditure dif_interest_rate dif_log_GDP dif_log_ERI
## [1,]      0.0001214474      0.0005117225 0.009689147 0.9896777
## [2,]      0.0061300247      0.0005051201 0.008446981 0.9611559
## [3,]      0.0040511146      0.0003297016 0.136043017 0.6231319
## [4,]      0.0038350220      0.0003209663 0.147224161 0.5892250
## [5,]      0.0040735559      0.0003191044 0.147700864 0.5853623
## [6,]      0.0040769037      0.0003095582 0.151869239 0.5677802
## [7,]      0.0047786958      0.0003089379 0.151649495 0.5665829
## [8,]      0.0048066659      0.0003088089 0.151704404 0.5663883
## [9,]      0.0048432612      0.0003086404 0.151715253 0.5656478
## [10,]     0.0048918980      0.0003085761 0.151786760 0.5655404
##
##      dif_log_revenue dif_log_deflator
## [1,]      0.00000000      0.0000000000
## [2,]      0.02358203      0.0001799623
## [3,]      0.12175699      0.1146872907
## [4,]      0.13492319      0.1244716418
## [5,]      0.13552042      0.1270237685
## [6,]      0.13333630      0.1426277604
## [7,]      0.13307694      0.1436030402
## [8,]      0.13314497      0.1436468112
## [9,]      0.13303207      0.1444529718
## [10,]     0.13304906      0.1444233499
##
## $dif_log_revenue
##      dif_log_expenditure dif_interest_rate dif_log_GDP dif_log_ERI
## [1,]      0.02064899      1.332018e-05 0.01653219 0.03006592
## [2,]      0.01749554      1.007309e-04 0.13037625 0.03238610
## [3,]      0.01644586      9.314392e-05 0.12138388 0.03959088
## [4,]      0.01559208      9.254635e-05 0.11602831 0.06236377
## [5,]      0.01636743      9.158856e-05 0.11520976 0.06829425
## [6,]      0.01636471      8.851605e-05 0.12452924 0.06852313
## [7,]      0.01633496      9.164862e-05 0.12563412 0.06810527
## [8,]      0.01632811      9.164373e-05 0.12559398 0.06854892
## [9,]      0.01631624      9.163107e-05 0.12568179 0.06849219
## [10,]     0.01643571      9.160316e-05 0.12575708 0.06845832
##
##      dif_log_revenue dif_log_deflator

```

```

## [1,]      0.9327396      0.00000000
## [2,]      0.8068858      0.01275562
## [3,]      0.7439808      0.07850547
## [4,]      0.7013291      0.10459420
## [5,]      0.6942439      0.10579308
## [6,]      0.6827497      0.10774470
## [7,]      0.6801106      0.10972337
## [8,]      0.6797864      0.10965090
## [9,]      0.6792360      0.11018216
## [10,]     0.6791481      0.11010920
##
## $dif_log_deflator
##      dif_log_expenditure dif_interest_rate dif_log_GDP dif_log_ERI
## [1,]      0.001934955      1.426439e-05 0.006823120 0.008549524
## [2,]      0.009451197      1.506007e-05 0.006834852 0.024565633
## [3,]      0.008465043      2.549488e-05 0.059276927 0.022666315
## [4,]      0.008133138      2.867852e-05 0.055192489 0.027228683
## [5,]      0.008072134      2.936707e-05 0.057217539 0.027354044
## [6,]      0.008099795      3.123791e-05 0.060786450 0.026807925
## [7,]      0.008050763      3.212010e-05 0.061605275 0.026617003
## [8,]      0.008022363      3.204252e-05 0.062431044 0.026565658
## [9,]      0.008010703      3.217133e-05 0.062808635 0.026614435
## [10,]     0.008005217      3.216958e-05 0.062984551 0.026606757
##      dif_log_revenue dif_log_deflator
## [1,]      0.03540383      0.9472743
## [2,]      0.03963132      0.9195019
## [3,]      0.06200162      0.8475646
## [4,]      0.05766571      0.8517513
## [5,]      0.05753875      0.8497882
## [6,]      0.05649363      0.8477810
## [7,]      0.05661100      0.8470838
## [8,]      0.05643794      0.8465109
## [9,]      0.05636586      0.8461682
## [10,]     0.05633227      0.8460390
##
## $JB
##
## JB-Test (multivariate)
##
## data: Residuals of VAR object reduced_VAR
## Chi-squared = 12858, df = 12, p-value < 2.2e-16
##
##
## $Skewness
##
## Skewness only (multivariate)

```

```
##
## data: Residuals of VAR object reduced_VAR
## Chi-squared = 690.91, df = 6, p-value < 2.2e-16
##
##
## $Kurtosis
##
## Kurtosis only (multivariate)
##
## data: Residuals of VAR object reduced_VAR
## Chi-squared = 12167, df = 6, p-value < 2.2e-16
```

6.3 Chow test for Structural Breaks

Interested in assessing differences in the fiscal multipliers pre / post GFC

6.3.1 Chow Test

Chow test for parameter stability - Lutkepohl and Candelon (2000) highlight that when the sample size is low relative to the number of parameters, the Chow test may have distorted size. Given that we are considering a 6-dimensional VAR in this analysis, with a sample period of less than 40 years at a quarterly frequency, this issue is likely present in this analysis. Therefore we do not perform a traditional Chow test. The paper proposes as an alternative a bootstrap version of the test which they show to have more desirable finite sample properties.

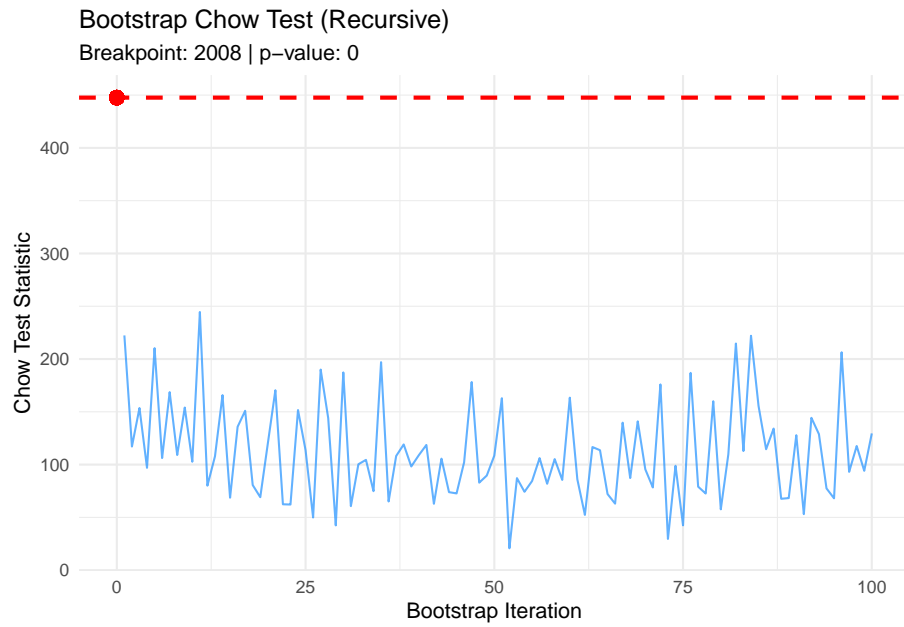
recursive residual-based bootstrap

The bootstrap test is implemented as follows:

- 1) Obtain the demeaned residuals
- 2) Generate bootstrap residuals by randomly drawing from the centered residuals with replacement.

Using sample-split test (asymptotic values) for chow: <https://www.econstor.eu/bitstream/10419/62173/1/723877181.pdf>

```
## Completed iteration 50
## Completed iteration 100
```



```
## [1] 447.6212
```

```
## [1] 0
```

```
## [1] 0
```

6.3.2 Pre GFC Shocks

6.3.3 Post GFC Shocks

```
## dif_log_GDP
##          -0.65
```

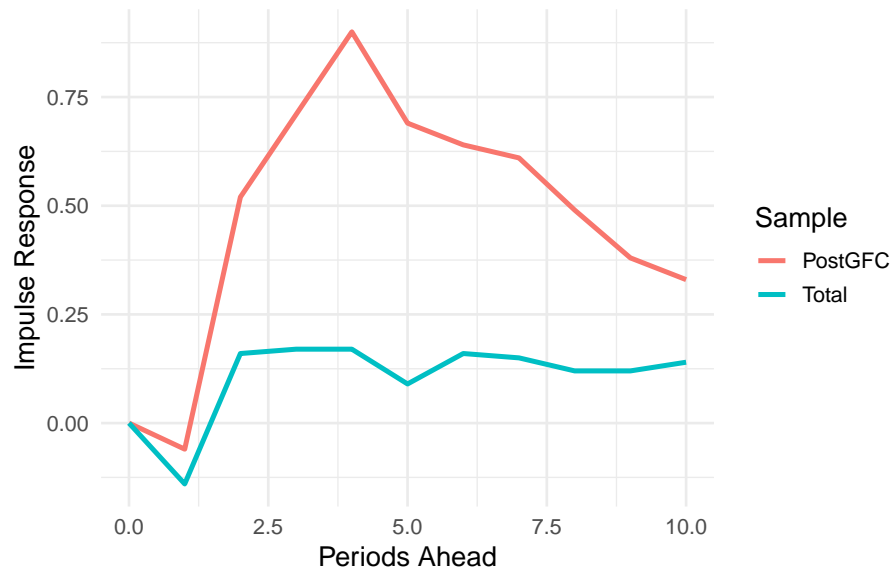
```
## dif_log_GDP
##          -0.65
```

Table 2: Cumulative IRFs by Horizon and Variable

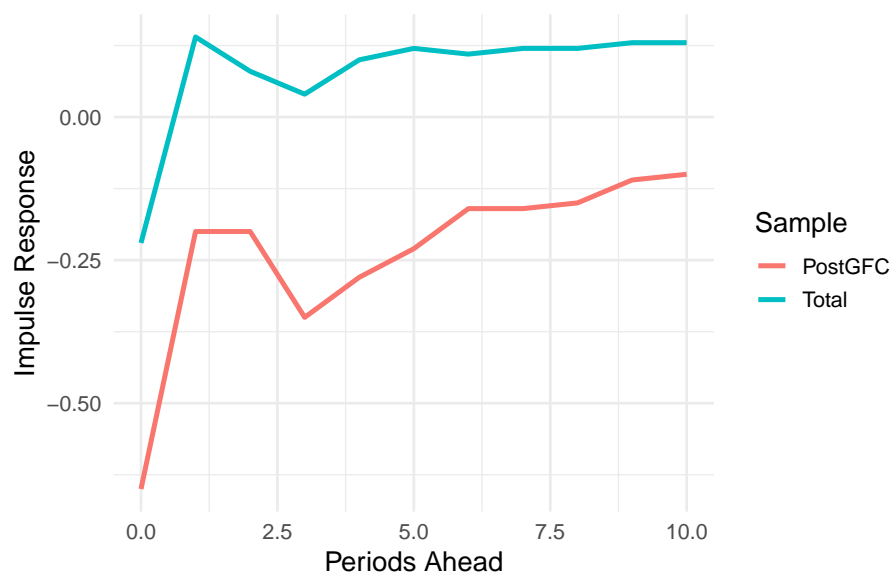
Period	Total_Expenditure	Total_Revenue	PostGFC_Expenditure	PostGFC_Revenue
0	-0.22	0.00	-0.65	0.00
1	0.14	-0.14	-0.20	-0.06

Period	Total_Expenditure	Total_Revenue	PostGFC_Expenditure	PostGFC_Revenue
2	0.08	0.16	-0.20	0.52
3	0.04	0.17	-0.35	0.71
4	0.10	0.17	-0.28	0.90
5	0.12	0.09	-0.23	0.69
6	0.11	0.16	-0.16	0.64
7	0.12	0.15	-0.16	0.61
8	0.12	0.12	-0.15	0.49
9	0.13	0.12	-0.11	0.38
10	0.13	0.14	-0.10	0.33

IRF: Revenue Response to Expenditure Shock



IRF: Expenditure Response to Revenue Shock



6.3.4 Post GFC Diagnostics

Ramey (2019) define fiscal multipliers: “change in output due to a change in spending or taxes”, and highlight the risk of ignoring fiscal foresight. Gechert (2017)

7 Robustness

7.1 Stability

7.2 Normality

7.3 Lag length

7.4 Identification

8 Discussion/ Policy Implications

8.1 Limitations

- potential heterogeneity within expenditure/ revenue

- debt sustainability (effects of policy on inflation and interest rate, changing real cost)

9 Conclusion

10 Bibliography

Cheung, Y.W. and Lai, K.S., 1993. FINITE-SAMPLE SIZES OF JOHANSEN'S LIKELIHOOD RATIO TESTS FOR. *Oxford Bulletin of Economics and statistics*, 55(3), p.1025.

Lütkepohl, H., Saikkonen, P. and Trenkler, C., 2001. Maximum eigenvalue versus trace tests for the cointegrating rank of a VAR process. *The Econometrics Journal*, 4(2), pp.287-310.

References

- [1] O.J. Blanchard and D. Leigh. "Growth forecast errors and fiscal multipliers". In: *American Economic Review* 103.3 (2013), pp. 117–120.
- [2] E. Ilzetzki, E.G. Mendoza, and C.A. Végh. "How big (small?) are fiscal multipliers?" In: *Journal of Monetary Economics* 60.2 (2013), pp. 239–254.
- [3] D. Sutherland, P. Hoeller, and R. Merola. "Fiscal consolidation: How much, how fast and by what means?" In: (2012).
- [4] T. Warmedinger, C.D. Checherita-Westphal, and P.H. De Cos. "Fiscal multipliers and beyond". In: *ECB Occasional Paper* 162 (2015).

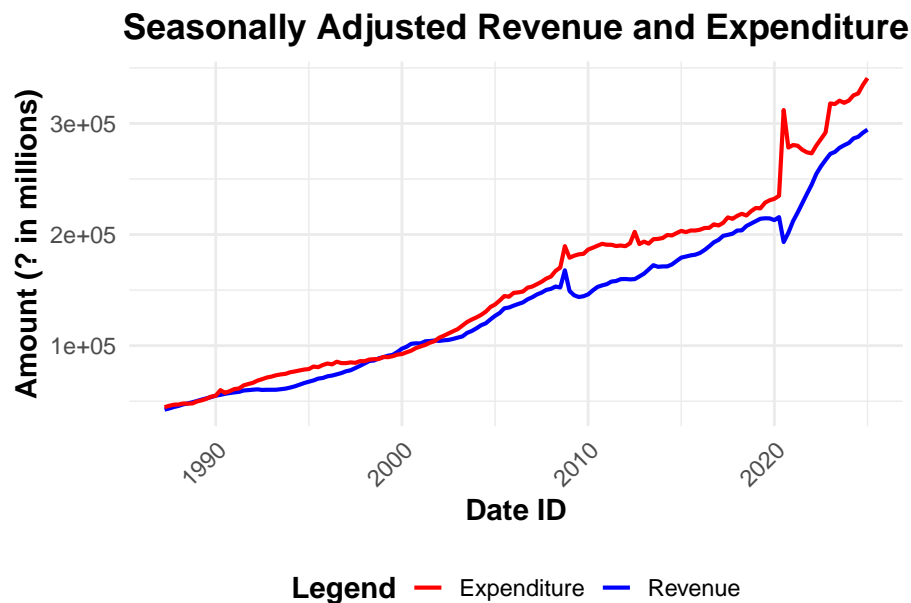
11 Data Appendix

11.1 Data Sources

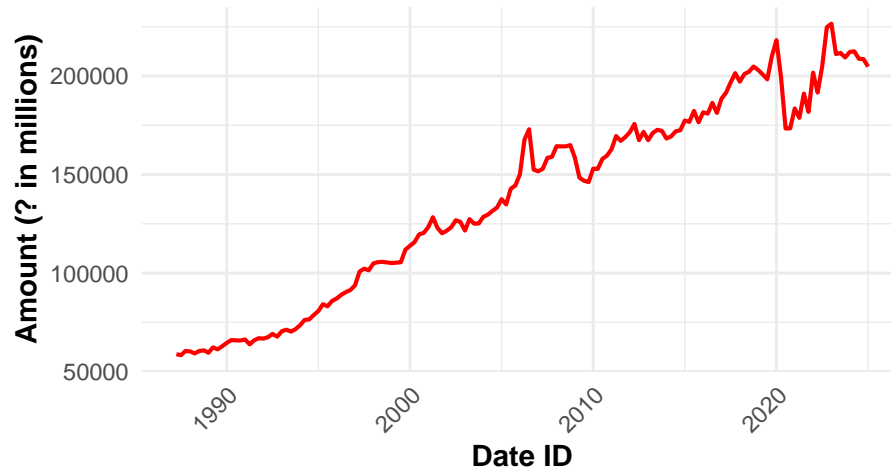
- Fiscal Variables (not seasonally adjusted):
<https://www.ons.gov.uk/economy/governmentpublicsectorandtaxes/publicspending/datasets/esatable25quarterlynonfinancialaccountsofgeneralgovernment>
- UK Exports (seasonally Adjusted, %):
<https://fred.stlouisfed.org/series/XTEXVA01GBQ188S>
- UK Exports (seasonally Adjusted, £millions):
<https://fred.stlouisfed.org/series/NXRSAXDCGBQ>

- LFS (Pop aged 16-64):
<https://www.ons.gov.uk/employmentandlabourmarket/peopleinwork/employmentandemployeetypes/timeseries/lf2o/lms>
- GDP (SA) and deflator rate:
<https://www.ons.gov.uk/economy/grossdomesticproductgdp/bulletins/quarterlynationalaccounts/latest#data-on-gdp-quarterly-national-accounts>
- interest rate (SR. This is dates of changes to the policy rate. Have interpolated to get quarterly data):
<https://www.bankofengland.co.uk/monetary-policy/the-interest-rate-bank-rate>
- 3 month interest rate:
<https://fred.stlouisfed.org/series/IR3TIB01GBM156N>

11.2 Data Plots

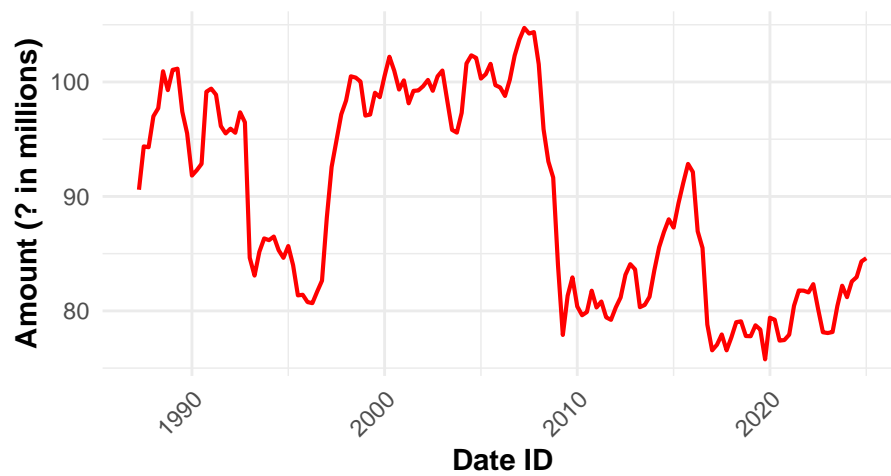


Exports Over Time

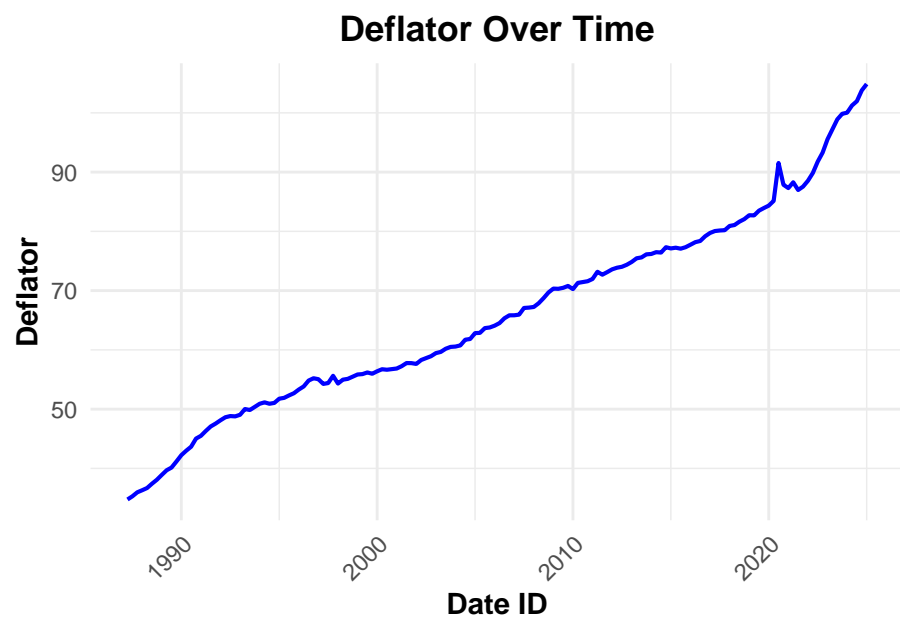
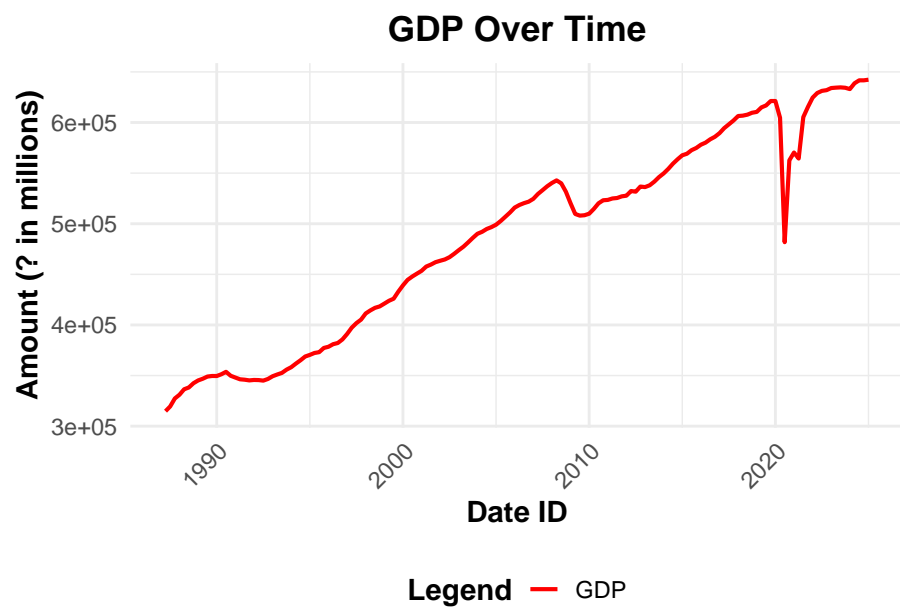


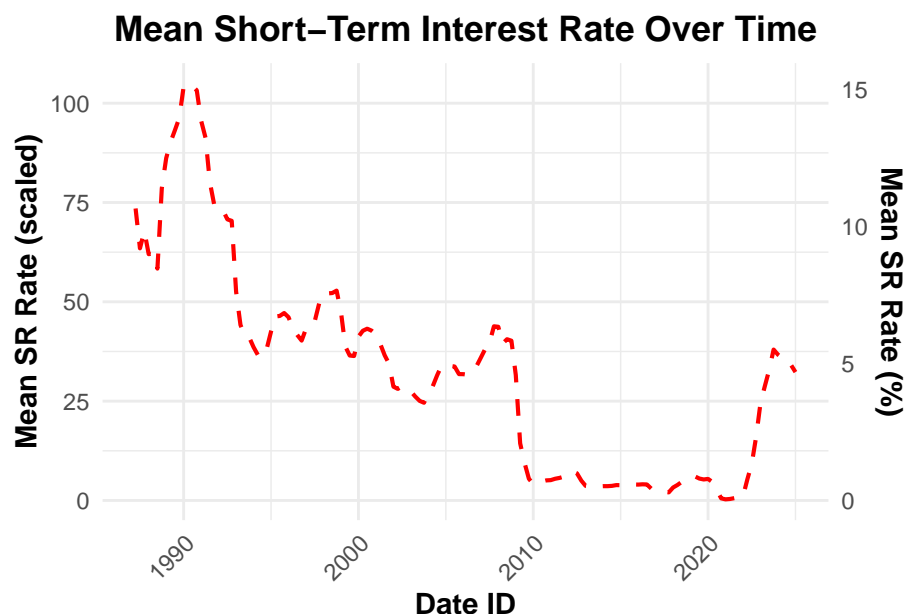
Legend — Exports

Exchange Rate Index Over Time



Legend — ERI





12 Technical Appendix

12.1 VECM Code/ Results

```
##
## #####
## # Johansen-Procedure #
## #####
##
## Test type: maximal eigenvalue statistic (lambda max) , with linear trend
##
## Eigenvalues (lambda):
## [1] 0.217415139 0.147896389 0.129451390 0.100436363 0.049461170 0.003638249
##
## Values of teststatistic and critical values of test:
##
##          test 10pct  5pct  1pct
## r <= 5 |   0.54   6.50   8.18 11.65
## r <= 4 |   7.51  12.91  14.90 19.19
## r <= 3 |  15.67  18.90  21.07 25.75
## r <= 2 |  20.52  24.78  27.14 32.14
## r <= 1 |  23.69  30.84  33.32 38.78
## r = 0 |  36.28  36.25  39.43 44.59
```

```

##
## Eigenvectors, normalised to first column:
## (These are the cointegration relations)
##
##          log_expenditure.l4 mean_SR_Rate.l4 log_GDP.l4 log_ERI.l4
## log_expenditure.l4          1.0000000          1.0000000          1.0000000          1.0000000
## mean_SR_Rate.l4          0.2475518          -0.8612854          0.1335571          0.01524865
## log_GDP.l4          -4.8496255          -69.6526523          32.2728925          1.56515827
## log_ERI.l4          -6.7246527          -5.1910987          -6.7313704          0.09115364
## log_revenue.l4          7.1339296          44.8805450          -16.1426688          -1.98663605
## log_deflator.l4          -12.0385559          -48.1017843          6.2168996          0.89066270
##          log_revenue.l4 log_deflator.l4
## log_expenditure.l4          1.000000000          1.0000000
## mean_SR_Rate.l4          0.008820036          -0.1080448
## log_GDP.l4          0.976624228          32.3208527
## log_ERI.l4          0.398010324          -9.1161183
## log_revenue.l4          -0.725452699          -20.8943405
## log_deflator.l4          -1.324829833          3.5873395
##
## Weights W:
## (This is the loading matrix)
##
##          log_expenditure.l4 mean_SR_Rate.l4 log_GDP.l4 log_ERI.l4
## log_expenditure.d          -0.002040882          -0.0041718236          0.004062505          -0.0821484594
## mean_SR_Rate.d          -0.094262868          0.0694052354          0.074581239          -0.9268431363
## log_GDP.d          -0.001546084          0.0027382737          -0.009498974          0.0449624926
## log_ERI.d          0.001583154          0.0017157783          0.002163005          0.0001462418
## log_revenue.d          -0.003901635          0.0009839674          -0.004212126          0.0548073959
## log_deflator.d          0.006006020          -0.0005837983          0.002546326          -0.0153086374
##          log_revenue.l4 log_deflator.l4
## log_expenditure.d          0.005588010          3.713822e-04
## mean_SR_Rate.d          0.183018284          7.112554e-03
## log_GDP.d          -0.007451016          -2.556513e-06
## log_ERI.d          -0.062536507          2.744741e-04
## log_revenue.d          0.005746378          3.562735e-04
## log_deflator.d          0.007333019          9.651040e-05
##
## #####
## # Johansen-Procedure #
## #####
##
## Test type: trace statistic , with linear trend
##
## Eigenvalues (lambda):
## [1] 0.217415139 0.147896389 0.129451390 0.100436363 0.049461170 0.003638249

```



```

##
## Values of teststatistic and critical values of test:
##
##          test 10pct  5pct   1pct
## r <= 5 |    0.54  6.50  8.18 11.65
## r <= 4 |    8.05 15.66 17.95 23.52
## r <= 3 |   23.71 28.71 31.52 37.22
## r <= 2 |   44.23 45.23 48.28 55.43
## r <= 1 |   67.92 66.49 70.60 78.87
## r = 0  | 104.20 85.18 90.39 104.20
##
## Eigenvectors, normalised to first column:
## (These are the cointegration relations)
##
##          log_expenditure.l4 mean_SR_Rate.l4 log_GDP.l4 log_ERI.l4
## log_expenditure.l4          1.0000000          1.0000000  1.0000000  1.0000000
## mean_SR_Rate.l4          0.2475518          -0.8612854  0.1335571  0.01524865
## log_GDP.l4          -4.8496255          -69.6526523  32.2728925  1.56515827
## log_ERI.l4          -6.7246527          -5.1910987  -6.7313704  0.09115364
## log_revenue.l4          7.1339296          44.8805450 -16.1426688 -1.98663605
## log_deflator.l4          -12.0385559          -48.1017843   6.2168996  0.89066270
##          log_revenue.l4 log_deflator.l4
## log_expenditure.l4          1.000000000          1.0000000
## mean_SR_Rate.l4          0.008820036          -0.1080448
## log_GDP.l4          0.976624228          32.3208527
## log_ERI.l4          0.398010324          -9.1161183
## log_revenue.l4          -0.725452699          -20.8943405
## log_deflator.l4          -1.324829833           3.5873395
##
## Weights W:
## (This is the loading matrix)
##
##          log_expenditure.l4 mean_SR_Rate.l4 log_GDP.l4 log_ERI.l4
## log_expenditure.d          -0.002040882          -0.0041718236  0.004062505 -0.0821484594
## mean_SR_Rate.d          -0.094262868          0.0694052354  0.074581239 -0.9268431363
## log_GDP.d          -0.001546084          0.0027382737 -0.009498974  0.0449624926
## log_ERI.d          0.001583154          0.0017157783  0.002163005  0.0001462418
## log_revenue.d          -0.003901635          0.0009839674 -0.004212126  0.0548073959
## log_deflator.d          0.006006020          -0.0005837983  0.002546326 -0.0153086374
##          log_revenue.l4 log_deflator.l4
## log_expenditure.d          0.005588010          3.713822e-04
## mean_SR_Rate.d          0.183018284          7.112554e-03
## log_GDP.d          -0.007451016          -2.556513e-06
## log_ERI.d          -0.062536507          2.744741e-04
## log_revenue.d          0.005746378          3.562735e-04
## log_deflator.d          0.007333019          9.651040e-05

```

```

##
## Coefficient matrix of lagged endogenous variables:
##
## A1:
##      log_expenditure.l1 mean_SR_Rate.l1 log_GDP.l1 log_ERI.l1
## log_expenditure      0.41636587      0.008657320 -0.26343648 -0.23871127
## mean_SR_Rate        -2.49494804      1.371479375  1.60128927  3.54054205
## log_GDP              0.32754046      0.002606730  1.05329908  0.11653213
## log_ERI             -0.10581266      0.002233156 -0.01297732  1.34600907
## log_revenue          0.07067758      0.007013725  0.40633349  0.07821876
## log_deflator        -0.04243531      0.002395629 -0.01860357 -0.12608423
##      log_revenue.l1 log_deflator.l1
## log_expenditure      0.1365826      0.49135673
## mean_SR_Rate         3.0428316      14.88586988
## log_GDP              -0.1752373     -0.11219143
## log_ERI             -0.1578161     -0.04745492
## log_revenue          0.6721756      0.28567039
## log_deflator         0.1158639      0.83190109
##
##
## A2:
##      log_expenditure.l2 mean_SR_Rate.l2 log_GDP.l2 log_ERI.l2
## log_expenditure      0.58145787     -0.011208795  0.5221245  0.4115522
## mean_SR_Rate         3.12626016     -0.271791808  5.0017738 -6.0529095
## log_GDP              -0.20993790     -0.006198375 -0.1233815 -0.2574373
## log_ERI              0.19503651     -0.007845216  0.5155834 -0.4473172
## log_revenue          -0.01587097     -0.012842288 -0.1491301 -0.1553034
## log_deflator         0.08659207     -0.001114785  0.1784110  0.1605157
##      log_revenue.l2 log_deflator.l2
## log_expenditure     -0.4417541      0.03931063
## mean_SR_Rate        -6.8593475     -7.28766381
## log_GDP              0.3759424     -0.03515733
## log_ERI             -0.1699028      0.47255237
## log_revenue          0.3990891      0.18995242
## log_deflator        -0.1926558      0.16850689
##
##
## A3:
##      log_expenditure.l3 mean_SR_Rate.l3 log_GDP.l3 log_ERI.l3
## log_expenditure      0.18134054      0.0016479441  0.136618036 -0.28187363
## mean_SR_Rate         -4.76043623     -0.2311996945 -13.516074516  5.54857449
## log_GDP              -0.16529249     -0.0009250504 -0.009612645  0.38618547
## log_ERI             -0.18832338      0.0064421792 -0.701478297  0.17322198
## log_revenue          -0.10624444      0.0028323025 -0.351965764  0.36761248
## log_deflator        -0.02679324      0.0003323898 -0.086380905 -0.09116484
##      log_revenue.l3 log_deflator.l3

```

```

## log_expenditure      0.11768226      -0.22889078
## mean_SR_Rate         9.60368984     -13.37039289
## log_GDP              -0.05342599     -0.02380320
## log_ERI              0.56292979     -0.62376686
## log_revenue          -0.02771123     -0.51527165
## log_deflator         0.07698485      0.07522735
##
##
## A4:
##           log_expenditure.l4 mean_SR_Rate.l4 log_GDP.l4 log_ERI.l4
## log_expenditure      -0.18120516      0.0003983068 -0.38540853  0.12275688
## mean_SR_Rate         4.03486125      0.1081771862  7.37015103 -2.40232197
## log_GDP              0.04614385      0.0041339600  0.08719302 -0.23488347
## log_ERI              0.10068269     -0.0004382070  0.19119456 -0.08256001
## log_revenue          0.04753621      0.0020304044  0.11368382 -0.26429068
## log_deflator         -0.01135751     -0.0001264335 -0.10255345  0.01634494
##           log_revenue.l4 log_deflator.l4
## log_expenditure      0.17292976     -0.27720731
## mean_SR_Rate         -6.45963860      6.90697564
## log_GDP              -0.15830885      0.18976458
## log_ERI              -0.22391682      0.17961052
## log_revenue          -0.07138742      0.08661888
## log_deflator         0.04265358     -0.14793913
##
##
## Coefficient matrix of deterministic regressor(s).
##
##           constant
## log_expenditure  0.0082761812
## mean_SR_Rate    -0.2985016017
## log_GDP         -0.0023718540
## log_ERI         0.0042222188
## log_revenue     -0.0005221706
## log_deflator    0.0171513474

## [1] "varest"

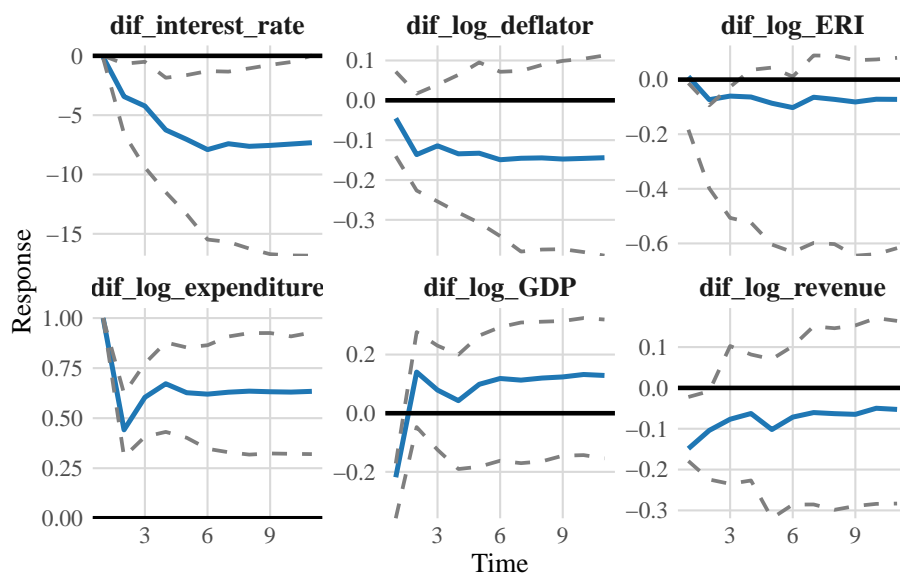
## [1] "vec2var"

```

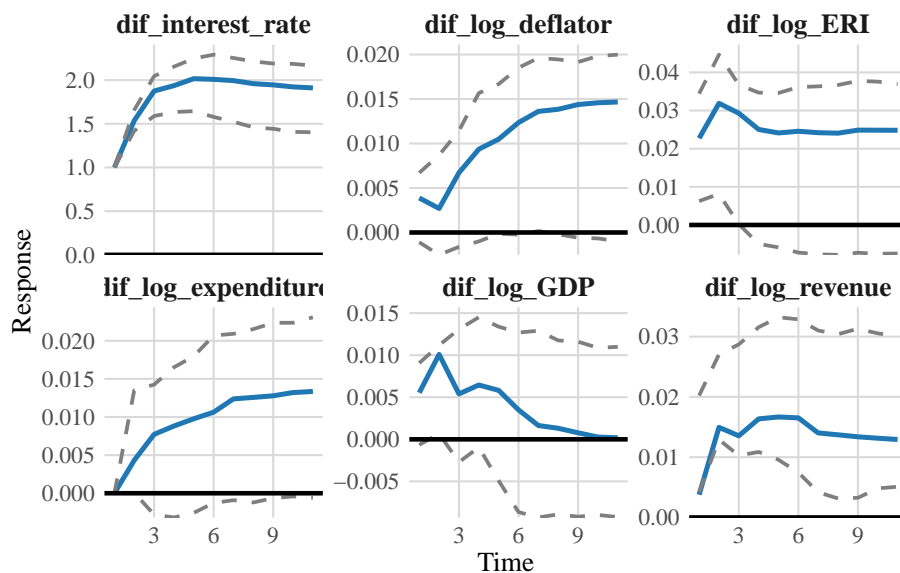
12.2 IRFs

12.2.1 Total Period:

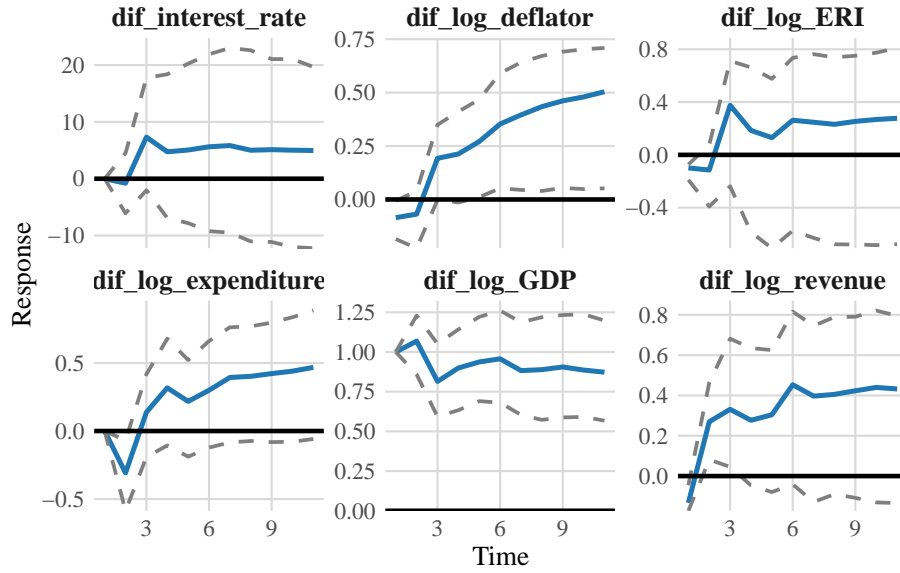
Impulse Response for Shock: dif_log_expenditure



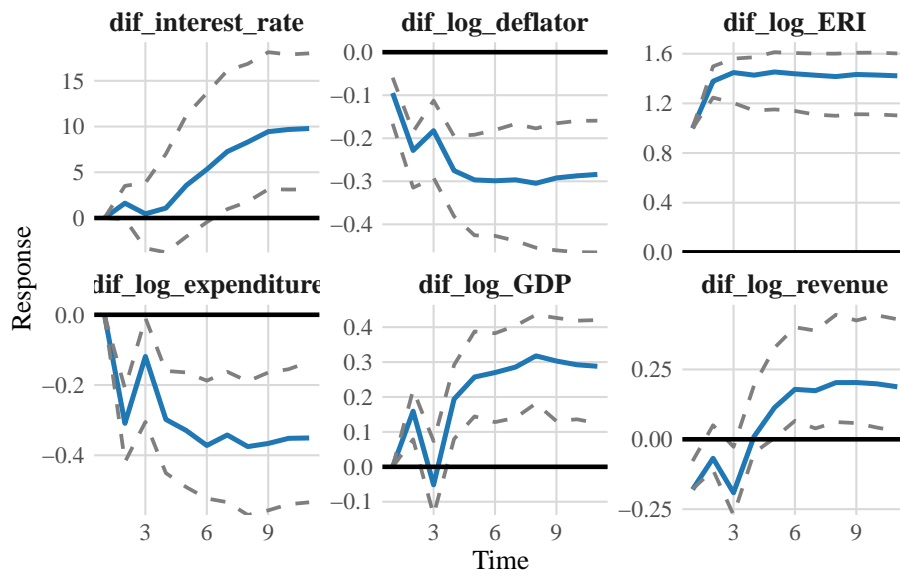
Impulse Response for Shock: dif_interest_rate



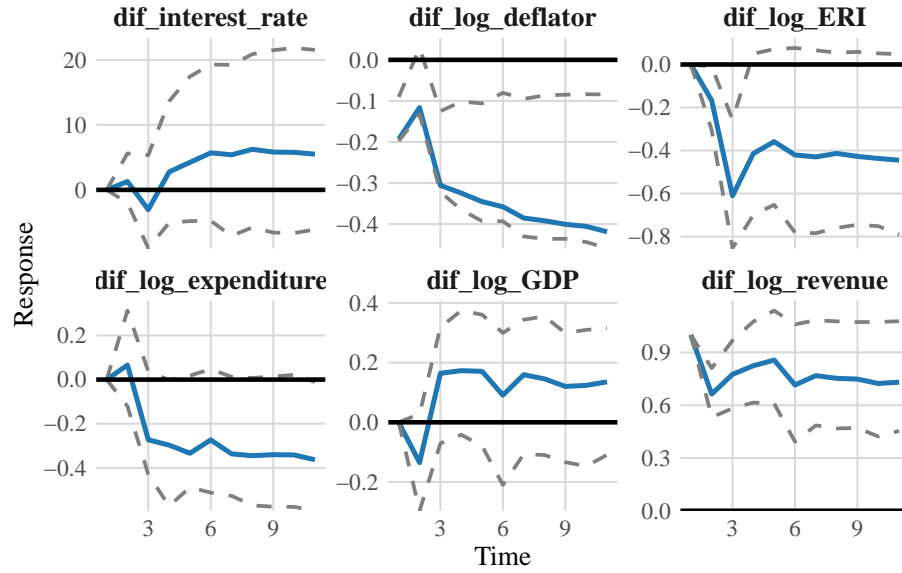
Impulse Response for Shock: dif_log_GDP



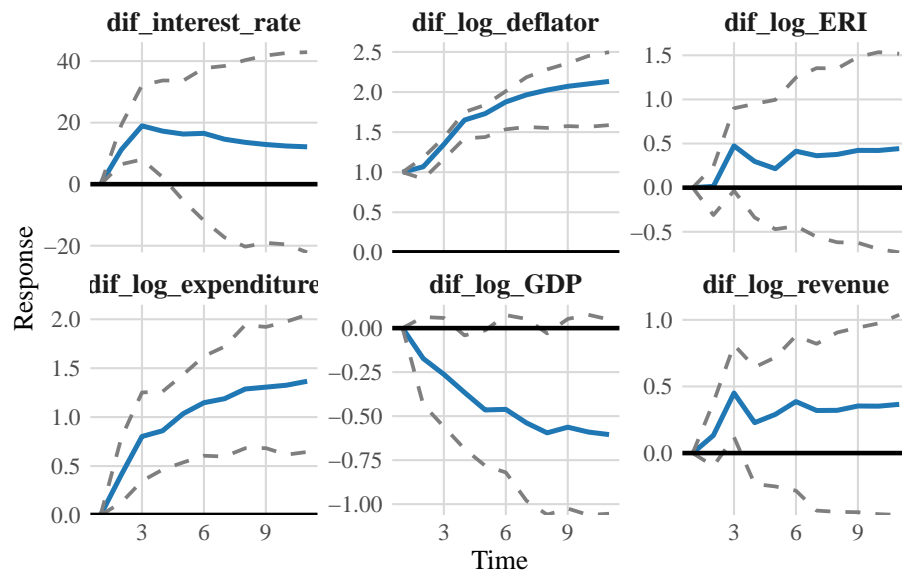
Impulse Response for Shock: dif_log_ERI



Impulse Response for Shock: dif_log_revenue



Impulse Response for Shock: dif_log_deflator



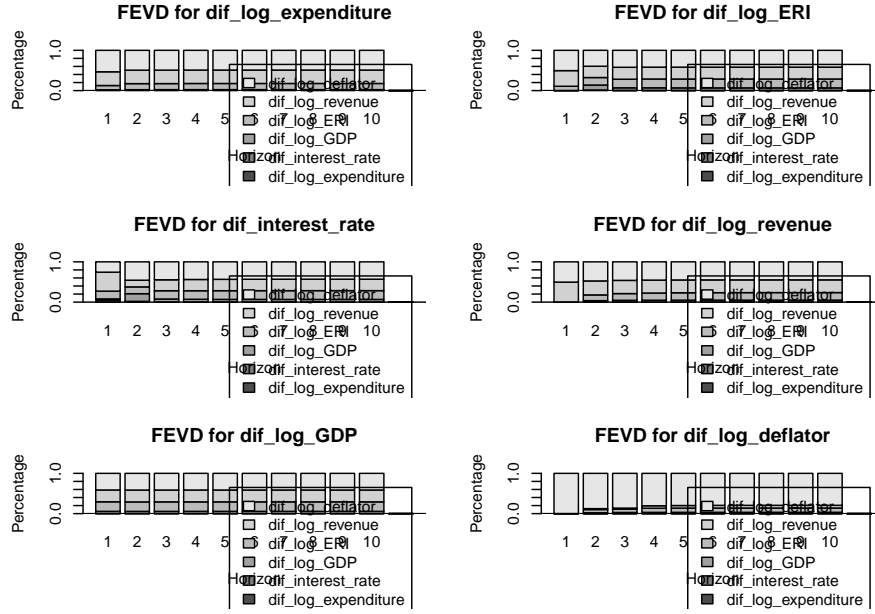
12.3 FEVDs post GFC:

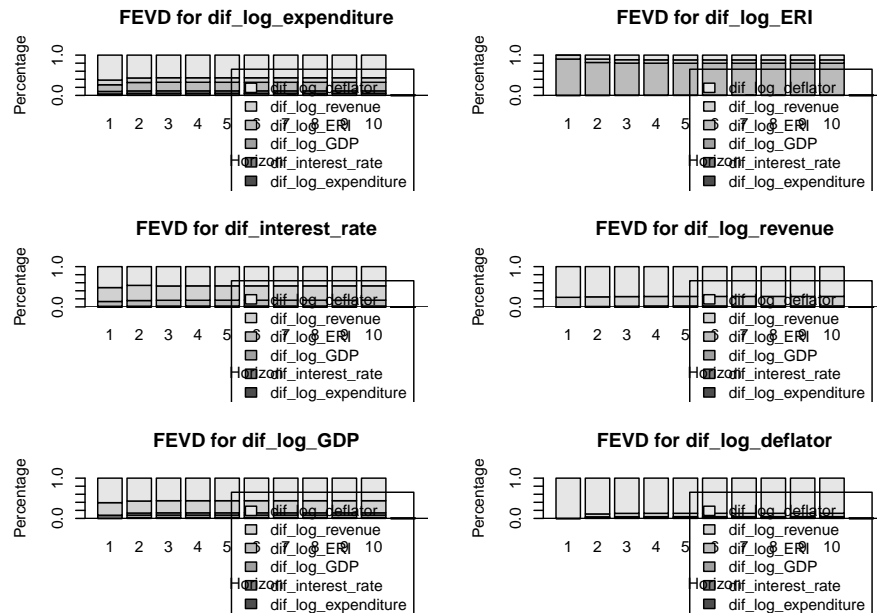
Table 3: Mean Values of Differenced Variables

Variable	Mean
dif_log_expenditure	0.01347
dif_interest_rate	-0.03960
dif_log_ERI	-0.00045
dif_log_GDP	0.00472
dif_log_revenue	0.01283
dif_log_deflator	0.00731

[1] 0.64023181 0.46388693 0.41259061 0.21210529 0.21210529 0.04085178

[1] 0.6784882 0.4679224 0.3903112 0.3903112 0.3438124 0.1635484





```
# library(knitr)
# library(stargazer)
# library(cliplr)
#library(kableExtra)

library(ggplot2)
library(knitr)
library(ivreg)
library(ggdag)
library(data.table)
library(dplyr)
library(tidyr)
library(stargazer)
library(cliplr)
library(tibble)
library(lubridate)
# install.packages("seasonal")
library(seasonal)

lapply(c("ggplot2", "dplyr", "data.table", "lubridate", "janitor", "broom", "tibble", "tidyr"),
       require, character.only = TRUE)
```



```

# knitr::opts_chunk$set(echo = FALSE)
knitr::opts_chunk$set(echo = FALSE, warning = FALSE, message = FALSE)

df <- fread("D:/Samid work/University/KCL - Econ and Policy/Dissertation/Data/GDP.csv", skip

# Filter the data frame to exclude rows where the column 'title' matches any of the specific

filtered_df <- df %>%
  # Keep only the quarterly data
  filter(nchar(CDID) == 7 & substr(CDID, 6, 6) == "Q") %>%
  # Select relevant columns and rename them
  dplyr::select(CDID, Deflator = LGGG, GDP = ABMI) %>%
  # Create new columns and convert types
  mutate(
    Year = as.numeric(substr(CDID, 1, 4)),
    Quarter = substr(CDID, 6, 7),
    Q = as.numeric(substr(CDID, 7, 7)),
    Deflator = as.numeric(Deflator),
    GDP = as.numeric(GDP)
  ) %>%
  # Filter by year (can modify for testing)
  filter(Year >= 1987)

fiscal_raw <- fread("D:/Samid work/University/KCL - Econ and Policy/Dissertation/Data/Fiscal

fiscal_proc <- fiscal_raw %>%
  dplyr::select(Date_ID = Transaction, Revenue = OTR, Expenditure = OTE) %>%
  subset(Date_ID != "Dataset identifier code" & Date_ID != "Identifier") %>%
  mutate(Year = as.numeric(gsub("\\D", "", Date_ID)),
    Period = gsub("\\d{4}", "", Date_ID)) %>%
  mutate(
    Q = case_when(
      Period == "Jan to Mar " ~ 1,
      Period == "Apr to Jun " ~ 2,
      Period == "Jul to Sep " ~ 3,
      Period == "Oct to Dec " ~ 4
    ),
    Unique_Period = Year +(Q/4)
  )

```

```

) %>%
  # Convert to numeric and multiply by 1 million so values as these will later be made into
  mutate(Revenue = as.numeric(gsub(",", "", Revenue) ),
         Expenditure = as.numeric(gsub(",", "", Expenditure )))

# join GDP deflator and GDP data

population <- fread("D:/Samid work/University/KCL - Econ and Policy/Dissertation/Data/Popula
                  skip = 4,
                  header = TRUE) %>%
subset(`Country Name` == "United Kingdom") %>%
t() %>%
as.data.frame() %>%
rownames_to_column(var = "Year") %>%
rename(Population = V1 ) %>%
filter(grepl("^\\d{4}$", Year)) %>%
mutate(Year = as.numeric(Year),
       Population = as.numeric(Population))

Interest_SR <- fread("D:/Samid work/University/KCL - Econ and Policy/Dissertation/Data/3 mor
mutate(Date = dmy(Date),
       month = month(Date),
       Year = year(Date),
       Q = case_when(
         month %in% 1:3 ~ 1,
         month %in% 4:6 ~ 2,
         month %in% 7:9 ~ 3,
         month %in% 10:12 ~ 4
       )) %>%
group_by(Year, Q) %>%
summarize(mean_SR_Rate = mean(Rate, na.rm = TRUE))

SONIA <- fread("D:/Samid work/University/KCL - Econ and Policy/Dissertation/Data/Bank of Eng
mutate(Date = dmy(Date),
       month = month(Date),
       Year = year(Date),
       Q = case_when(
         month %in% 1:3 ~ 1,
         month %in% 4:6 ~ 2,
         month %in% 7:9 ~ 3,
         month %in% 10:12 ~ 4
       )) %>%
group_by(Year, Q) %>%

```

```

summarize(mean_SONIA = mean(SONIA, na.rm = TRUE))

Policy_Rate <- fread("D:/Samid work/University/KCL - Econ and Policy/Dissertation/Data/Policy_Rate.csv")
mutate(Date = parse_date_time(`Date Changed`, orders = "dmy"),
       Q = quarter(Date),
       Year = year(Date)) %>%
group_by(Year, Q) %>%
summarise(mean_SR_Rate = mean(Rate, na.rm = TRUE), .groups = "drop") %>%
complete(Year = full_seq(Year, 1), Q = 1:4) %>% # Ensure all Year-Quarter combinations
arrange(Year, Q) %>%
fill(mean_SR_Rate, .direction = "down") # Fill missing rates by propagating the previous

Exports <- fread("D:/Samid work/University/KCL - Econ and Policy/Dissertation/Data/Exports.csv")
mutate(Date = dmy(Month),
       month = month(Month),
       Year = year(dmy(Month)),
       Q = case_when(
         month %in% 1:3 ~ 1,
         month %in% 4:6 ~ 2,
         month %in% 7:9 ~ 3,
         month %in% 10:12 ~ 4
       )) %>%
group_by(Year, Q) %>%
summarize(Exports = sum(Exports, na.rm = TRUE))

ERI <- fread("D:/Samid work/University/KCL - Econ and Policy/Dissertation/Data/Boe ERI.csv")
mutate(
  month = month(Date),
  Year = year(dmy(Date)),
  Q = case_when(
    month %in% 1:3 ~ 1,
    month %in% 4:6 ~ 2,
    month %in% 7:9 ~ 3,
    month %in% 10:12 ~ 4
  )) %>%
rename(ERI = Value)

data <- fiscal_proc %>%
left_join(filtered_df, by = c("Q" = "Q", "Year" = "Year")) %>%
left_join(Interest_SR, by = c("Q" = "Q", "Year" = "Year")) %>%
left_join(Exports, by = c("Q" = "Q", "Year" = "Year")) %>%
left_join(population, by = c("Year" = "Year")) %>%
left_join(ERI, by = c("Q" = "Q", "Year" = "Year")) %>%
# Convert variables to per capita, note revenue, expenditure, and GDP are in £ million so n

```

```

mutate(RevenuePerCapita = (Revenue *10^6) /Population,
       ExpenditurePerCapita = (Expenditure *10^6) /Population,
       GDPPerCapita = (GDP *10^6) /Population) %>%

# Seasonal Adjustment of data using X-13ARIMA-SEATS
mutate(Revenue_SA = final(seas(ts(Revenue, start = min(Year), frequency = 4))),
       Expenditure_SA = final(seas(ts(Expenditure, start = min(Year), frequency = 4))),
       GDP_SA = final(seas(ts(GDP, start = min(Year), frequency = 4)))) %>%
# convert back to numeric
mutate(
  Revenue_SA = as.numeric(Revenue_SA),
  Expenditure_SA = as.numeric(Expenditure_SA),
  GDP_SA = as.numeric(GDP_SA)
) %>%

# Convert variables (except interest rate) to logs
# Note multiplying by 10^6 for variables that are defined in £millions
mutate(log_revenue = log(Revenue_SA *10^6),
       log_expenditure = log(Expenditure_SA *10^6),
       log_GDP = log(GDP_SA *10^6),
       log_deflator = log(Deflator),
       log_ERI = log(ERI),
       log_exports = log(Exports *10^6)) %>%
mutate(
  dif_log_revenue = log_revenue - lag(log_revenue),
  dif_log_expenditure = log_expenditure - lag(log_expenditure),
  dif_log_GDP = log_GDP - lag(log_GDP),
  dif_log_deflator = log_deflator - lag(log_deflator),
  dif_log_ERI = log_ERI - lag(log_ERI),
  dif_interest_rate = mean_SR_Rate - lag(mean_SR_Rate),
  dif_log_exports = log_exports - lag(log_exports)
)

model_data2 <- data %>%
  dplyr::select(Year, CDID, log_expenditure, mean_SR_Rate, log_GDP, log_ERI, log_revenue, 1

# model_data2 <- data %>%
#   dplyr::select(Year, CDID, log_expenditure, mean_SR_Rate, log_GDP, log_exports, log_revenue

# model_data <- data %>%
#   dplyr::select(Year, CDID, log_expenditure, mean_SR_Rate, log_exports, log_GDP, log_revenue

# model_data <- data %>%

```

```

#   dplyr::select(Year, CDID, dif_log_expenditure, dif_interest_rate, dif_log_exports, dif_log_GDP, dif_log_ERI, dif_log_GDP)

model_data <- data %>%
dplyr::select(Year, CDID, dif_log_expenditure, dif_interest_rate, dif_log_GDP, dif_log_ERI, dif_log_GDP)

# Adding Exports

clean_data <- na.omit(model_data)

clean_data2 <- na.omit(model_data2)

tmp <- clean_data[, -c(1,2)]

Optimallag <- VARselect(clean_data[, -c(1,2)], lag.max = 5, type = "const")
Optimallag2 <- VARselect(clean_data2[, -c(1,2)], lag.max = 5, type = "const")

# Optimallag2$selection
# # Optimallag$criteria
# Optimallag$selection
# Optimallag$criteria
# Optimallag2$criteria

# library(vars)

reduced_VAR <- VAR(clean_data[, -c(1,2)], p = 3, type = "const")
# reduced_VAR2 <- VAR(clean_data2[, -c(1,2)], p = 4, type = "both")
# reduced_VAR <- VAR(clean_data[, -1], p = 4)

# roots(reduced_VAR)
# roots(reduced_VAR2)

# summary(reduced_VAR)

# Summary reports the roots of the polynomial.

```

```

# Define the 5 dimensional lower triangular matrix, A

# Recover structural VAR using Cholesky decomposition

# Amat <- matrix(c(1, 0, 0, 0, 0,   # Recursive ordering
#                 NA, 1, 0, 0, 0,
#                 NA, NA, 1, 0, 0,
#                 NA, NA, NA, 1, 0,
#                 NA, NA, NA, NA, 1),
#               nrow = 5, byrow = TRUE)

# Variables are already ordered per the recursive identification strategy. Thus create a lower triangular matrix
k <- ncol(clean_data[, -c(1,2)])
Amat <- diag(1, k)
Amat[lower.tri(Amat)] <- NA
# print(Amat)

svar_model <- SVAR(reduced_VAR, Amat = Amat, estmethod = "direct")
# svar_model2 <- SVAR(reduced_VAR2, Amat = Amat, estmethod = "direct")
# ?SVAR()

structural_shocks <- residuals(svar_model)

# svar_model

irf_result <- irf(svar_model, n.ahead = 10, ci = 0.68, boot = 5000, cumulative = TRUE) # Forecast horizons
# irf_result2 <- irf(svar_model2, n.ahead = 10, ci = 0.68, boot = 5000, cumulative = FALSE)
# irf_result <- irf(svar_model, n.ahead = 10, ci = 0.68, boot = 5000, cumulative = TRUE) #
# plot(irf_result)
# Visualize IRFs

FEVD_result <- fevd(svar_model, n.ahead = 10) # Forecast horizons
# FEVD_result2 <- fevd(svar_model2, n.ahead = 10) # Forecast horizons
plot(FEVD_result)
# plot(FEVD_result2)

# Post Covid:

```

```

# Extract IRFs
exp_irf <- round(irf_result$irf$diff_log_expenditure[, 3],2)
rev_irf <- round(irf_result$irf$diff_log_revenue[, 3],2)


# Create horizon index from 0
horizon <- 0:(length(exp_irf) - 1)

# Build data frame
output_table <- data.frame(
  Horizon = rep(horizon, 2),
  Variable = rep(c("diff_log_expenditure", "diff_log_revenue"), each = length(horizon)),
  IRF = round(c(exp_irf, rev_irf), 5)
)

# Display table
kable(output_table, align = "lrrr", caption = "Cumulative IRF by Horizon and Variable")


FEVD_result


# structural_shocks <- residuals(svar_model)
# irf_result <- irf(svar_model, n.ahead = 10) # Forecast horizons
# plot(irf_result) # Visualize IRFs


normality.test(reduced_VAR)


# roots(reduced_VAR)

library(vars)
library(boot)

```

```

# --- Step 1: Define Restricted Model ---
# reduced_VAR <- VAR(clean_data[,-c(1,2)], p = 3, type = "const")
# --- Step 1: Define Parameters ---
break_year <- 2008
p <- 1
Rep <- 100
T_break <- which((clean_data$Year == break_year))[1] # First observation of break year

# --- Step 2: Estimate VAR & Prepare Data ---
Y <- clean_data[, -c(1, 2)] # Drop Year + Quarter columns
var_orig <- VAR(Y, p = p, type = "const")
resids <- residuals(var_orig)
centered_resids <- scale(resids, scale = FALSE) # Centered residuals
Y_vals <- as.matrix(Y)
presample <- Y_vals[1:p, ]
coef_matrix <- Bcoef(var_orig)
A <- as.matrix(coef_matrix[, -ncol(coef_matrix)])
c <- coef_matrix[, ncol(coef_matrix)]

# --- Step 3: Define Chow Function (Lambda_BP) ---
chow_stat <- function(Y, T_break, p) {
  tryCatch({
    T_full <- nrow(Y)

    # Restricted model: full sample
    model_restricted <- VAR(Y, p = p, type = "const")
    SSR_restricted <- sum(residuals(model_restricted)^2)

    # Unrestricted model: separate pre and post samples
    Y_pre <- Y[1:T_break, ]
    Y_post <- Y[(T_break + 1):T_full, ]

    model_pre <- VAR(Y_pre, p = p, type = "const")
    model_post <- VAR(Y_post, p = p, type = "const")

    SSR_unrestricted <- sum(residuals(model_pre)^2) + sum(residuals(model_post)^2)
    SSR_1 <- sum(residuals(model_pre)^2)
    SSR_tot <- sum(residuals(model_restricted)^2)

    # Extract log-likelihoods
    ll_restricted <- logLik(model_restricted)
    ll_unrestricted <- logLik(model_pre) + logLik(model_post)
  }, error = function(e) {
    return(NA)
  })
}

```



```

# Count number of coefficients per equation (excluding stats like SE, t)
k <- nrow(coef(reduced_VAR)[[1]])
n_eq <- length(coef(reduced_VAR)) # Number of equations

# Degrees of freedom = number of extra parameters in unrestricted models
df_chow <- k * n_eq

# Compute LR test statistic
lambda_BP <- -2 * (as.numeric(ll_restricted) - as.numeric(ll_unrestricted))

return(lambda_BP)
}, error = function(e) {
  message("Chow statistic failed: ", e$message)
  return(NA)
})
}

# --- Step 4: Bootstrap Simulation ---
set.seed(1234)
boot_stats <- numeric(Rep)

for (r in 1:Rep) {
  # FIX: Sample from available residuals only
  u_star <- centered_resids[sample(1:nrow(centered_resids), size = nrow(Y_vals), replace = T)]

  Y_boot <- matrix(NA, nrow = nrow(Y_vals), ncol = ncol(Y_vals),
    dimnames = list(NULL, colnames(Y_vals)))
  Y_boot[1:p, ] <- presample

  for (t in (p + 1):nrow(Y_vals)) {
    Y_lags <- as.vector(t(Y_boot[(t - p):(t - 1), ]))
    Y_boot[t, ] <- c + A %*% Y_lags + u_star[t, ]
  }

  boot_stats[r] <- if (anyNA(Y_boot)) NA else chow_stat(Y_boot, T_break, p)
  if (r %% 50 == 0) cat("Completed iteration", r, "\n")
}

boot_stats <- na.omit(boot_stats)

# --- Step 5: Actual Statistic & p-value ---
actual_stat <- chow_stat(Y_vals, T_break, p)
p_value_asymptotic <- 1 - pchisq(actual_stat, df_chow)
p_val <- mean(boot_stats > actual_stat)

```

```

# --- Step 6: Plot Results ---
if (length(boot_stats) > 0) {
  boot_df <- data.frame(
    Iteration = 1:length(boot_stats),
    Chow_Statistic = boot_stats
  )

  ggplot(boot_df, aes(x = Iteration, y = Chow_Statistic)) +
    geom_line(color = "dodgerblue", alpha = 0.7) +
    geom_hline(yintercept = actual_stat, color = "red", linetype = "dashed", linewidth = 1)
    geom_point(aes(x = 0, y = actual_stat), color = "red", size = 3) +
    labs(
      title = "Bootstrap Chow Test (Recursive)",
      subtitle = paste("Breakpoint:", break_year, "| p-value:", round(p_val, 4)),
      x = "Bootstrap Iteration",
      y = "Chow Test Statistic"
    ) +
    theme_minimal()
} else {
  warning("No valid bootstrap statistics available to plot.")
}

print(actual_stat)
print(p_val)

print(p_value_asymptotic)

# plot_irf_with_ci(irf_result_a_data)
# plot_irf_with_ci(irf_result_a_data2)

T_break <- which((clean_data$Year == break_year))[1] # First observation of break year
Y_post_GFC <- Y[(T_break + 1):nrow(Y), ]

# reduced_VAR <- VAR(clean_data[, -c(1,2)], p = 3, type = "const")

reduced_VAR_post_GFC <- VAR(Y_post_GFC, p = 3, type = "const")

svar_model_post_gfc <- SVAR(reduced_VAR_post_GFC, Amat = Amat, estmethod = "direct")

```

```

irf_result_post_gfc <- irf(svar_model_post_gfc, n.ahead = 10, ci = 0.68, boot = 5000, cumul

round(irf_result_post_gfc$irf$dif_log_expenditure[1, 3],2)
round(irf_result_post_gfc$irf$dif_log_expenditure[1, 3],2)

# Uncomment line below
# plot_irf_with_ci(irf_result_b_data)

# plot_irf_with_ci(irf_result_b_data2)

# normality.test(reduced_VAR2)
#
# irf_long2 <- process_irf(irf_result2)
#
#
# plot_irf(irf_long2)
#
# roots(reduced_VAR2)

exp_irf_post_GFC <- round(irf_result_post_gfc$irf$dif_log_expenditure[, 3],2)
rev_irf_post_GFC <- round(irf_result_post_gfc$irf$dif_log_revenue[, 3],2)

# Combine into one table
# Combine IRFs into one table
irf_combined <- cbind(
  Period = 0:(length(exp_irf) - 1),
  Total_Expenditure = exp_irf,
  Total_Revenue = rev_irf,
  PostGFC_Expenditure = exp_irf_post_GFC,
  PostGFC_Revenue = rev_irf_post_GFC
)

# kable(irf_combined, digits = 4, caption = "Impulse Response Comparison: Total Period vs P
#   kable_styling(full_width = FALSE, bootstrap_options = c("striped", "hover")) %>%
#   add_header_above(c(" " = 1, "Total Period" = 2, "Post-GFC" = 2))

kable(irf_combined, align = "lrrr", caption = "Cumulative IRFs by Horizon and Variable")

# Prepare data for Revenue plot

```

```

df_rev <- data.frame(
  Period = 0:(length(rev_irf) - 1),
  Total = rev_irf,
  PostGFC = rev_irf_post_GFC
) %>%
  pivot_longer(cols = -Period, names_to = "Sample", values_to = "IRF")

# Plot Revenue IRFs
ggplot(df_rev, aes(x = Period, y = IRF, color = Sample)) +
  geom_line(size = 1.2) +
  labs(
    title = "IRF: Revenue Response to Expenditure Shock",
    x = "Periods Ahead",
    y = "Impulse Response"
  ) +
  theme_minimal(base_size = 14) +
  theme(plot.title = element_text(face = "bold", hjust = 0.5))

# Prepare data for Expenditure plot
df_exp <- data.frame(
  Period = 0:(length(exp_irf) - 1),
  Total = exp_irf,
  PostGFC = exp_irf_post_GFC
) %>%
  pivot_longer(cols = -Period, names_to = "Sample", values_to = "IRF")

# Plot Expenditure IRFs
ggplot(df_exp, aes(x = Period, y = IRF, color = Sample)) +
  geom_line(size = 1.2) +
  labs(
    title = "IRF: Expenditure Response to Revenue Shock",
    x = "Periods Ahead",
    y = "Impulse Response"
  ) +
  theme_minimal(base_size = 14) +
  theme(plot.title = element_text(face = "bold", hjust = 0.5))

## Extract IRFs
# exp_irf_post_GFC <- round(irf_result_post_gfc$irf$dif_log_expenditure[, 3], 2)
# rev_irf_post_GFC <- round(irf_result_post_gfc$irf$dif_log_revenue[, 3], 2)
#
#
## Create horizon index from 0
## horizon <- 0:(length(exp_irf_post_GFC) - 1)
#

```

```

# # Build data frame
# output_table_post_GFC <- data.frame(
#   Horizon = rep(horizon, 2),
#   Variable = rep(c("dif_log_expenditure", "dif_log_revenue"), each = length(horizon)),
#   IRF = round(c(exp_irf_post_GFC, rev_irf_post_GFC), 5)
# )
#
# # Display table
# kable(output_table, align = "lrrr", caption = "Cumulative IRF by Horizon and Variable")
#
# kable(output_table_post_GFC, align = "lrrr", caption = "Post Covid: Cumulative IRF by Horizon and Variable")
#
# serial_test <- serial.test(reduced_VAR_b_data, lags.pt = 16, type = "PT.asymptotic")
#
#
# serial_test # Don't reject H0 of no autocorrelation
#
# normality_test <- normality.test(reduced_VAR_b_data)
# normality_test
#
#
# hetero_test <- arch.test(reduced_VAR_b_data, lags.multi = 5)
# hetero_test

ggplot(data, aes(x = Unique_Period)) +
  geom_line(aes(y = Revenue_SA, color = "Revenue"), size = 1) +
  geom_line(aes(y = Expenditure_SA, color = "Expenditure"), size = 1) +
  labs(
    x = "Date ID",
    y = "Amount (? in millions)",
    title = "Seasonally Adjusted Revenue and Expenditure",
    color = "Legend"
  ) +
  scale_color_manual(values = c("Revenue" = "blue", "Expenditure" = "red")) +
  theme_minimal(base_size = 15) +
  theme(
    axis.text.x = element_text(angle = 45, hjust = 1),
    plot.title = element_text(hjust = 0.5, face = "bold"),
    axis.title.x = element_text(face = "bold"),
    axis.title.y = element_text(face = "bold"),
    legend.position = "bottom",
    legend.title = element_text(face = "bold")
  )

```

```

ggplot(data, aes(x = Unique_Period)) +
  geom_line(aes(y = Exports, color = "Exports"), size = 1) +
  labs(
    x = "Date ID",
    y = "Amount (? in millions)",
    title = "Exports Over Time",
    color = "Legend"
  ) +
  scale_color_manual(values = c("Exports" = "red")) +
  theme_minimal(base_size = 15) +
  theme(
    axis.text.x = element_text(angle = 45, hjust = 1),
    plot.title = element_text(hjust = 0.5, face = "bold"),
    axis.title.x = element_text(face = "bold"),
    axis.title.y = element_text(face = "bold"),
    legend.position = "bottom",
    legend.title = element_text(face = "bold")
  )

```

```

ggplot(data, aes(x = Unique_Period)) +
  geom_line(aes(y = ERI, color = "ERI"), size = 1) +
  labs(
    x = "Date ID",
    y = "Amount (? in millions)",
    title = "Exchange Rate Index Over Time",
    color = "Legend"
  ) +
  scale_color_manual(values = c("ERI" = "red")) +
  theme_minimal(base_size = 15) +
  theme(
    axis.text.x = element_text(angle = 45, hjust = 1),
    plot.title = element_text(hjust = 0.5, face = "bold"),
    axis.title.x = element_text(face = "bold"),
    axis.title.y = element_text(face = "bold"),
    legend.position = "bottom",
    legend.title = element_text(face = "bold")
  )

```

```

ggplot(data, aes(x = Unique_Period)) +
  geom_line(aes(y = GDP, color = "GDP"), size = 1) +
  labs(

```

```

    x = "Date ID",
    y = "Amount (? in millions)",
    title = "GDP Over Time",
    color = "Legend"
  ) +
  scale_color_manual(values = c("GDP" = "red")) +
  theme_minimal(base_size = 15) +
  theme(
    axis.text.x = element_text(angle = 45, hjust = 1),
    plot.title = element_text(hjust = 0.5, face = "bold"),
    axis.title.x = element_text(face = "bold"),
    axis.title.y = element_text(face = "bold"),
    legend.position = "bottom",
    legend.title = element_text(face = "bold")
  )

ggplot(data, aes(x = Unique_Period)) +
  geom_line(aes(y = Deflator), color = "blue", size = 1) +
  labs(
    x = "Date ID",
    y = "Deflator",
    title = "Deflator Over Time"
  ) +
  theme_minimal(base_size = 15) +
  theme(
    axis.text.x = element_text(angle = 45, hjust = 1),
    plot.title = element_text(hjust = 0.5, face = "bold"),
    axis.title.x = element_text(face = "bold"),
    axis.title.y = element_text(face = "bold")
  )

# Define scaling factor
scale_factor2 <- max(data$Deflator, na.rm = TRUE) / max(data$mean_SR_Rate, na.rm = TRUE)

ggplot(data, aes(x = Unique_Period)) +
  geom_line(aes(y = mean_SR_Rate * scale_factor2), color = "red", size = 1, linetype = "dashed") +
  scale_y_continuous(
    name = "Mean SR Rate (scaled)",
    sec.axis = sec_axis(~ . / scale_factor2, name = "Mean SR Rate (%)")
  ) +
  labs(
    x = "Date ID",
    title = "Mean Short-Term Interest Rate Over Time"
  )

```

```

) +
theme_minimal(base_size = 15) +
theme(
  axis.text.x = element_text(angle = 45, hjust = 1),
  plot.title = element_text(hjust = 0.5, face = "bold"),
  axis.title.x = element_text(face = "bold"),
  axis.title.y = element_text(face = "bold")
)
# library(urca)

# urca::cajolst()

# vec2var
# Perform cointegration testing on the (log) levels variables
ca.jo_result_max_eig <- ca.jo(clean_data2[, -c(1,2)], type = "eigen", ecdet = "none", K = 4)

ca.jo_result_trace <- ca.jo(clean_data2[, -c(1,2)], type = "trace", ecdet = "none", K = 4)

ca.po_result <- ca.po(clean_data2[, -c(1,2)], demean = "const", type = "Pz")

# ca.lut_result <- cajolst(clean_data2[, -c(1,2)], trend = TRUE, K = 2, season = NULL)

summary(ca.jo_result_max_eig)
summary(ca.jo_result_trace)
# summary(ca.po_result)
# summary(ca.lut_result)

# ca.jo()

# vecm <- urca::cajools(ca.jo_result)

# plotres(ca.jo_result)

# vecm <- cajorls(ca.jo_result, r = 1)
johansen_test <- cajorls(ca.jo_result_max_eig, r = 1)

# johansen_test

vec2var_model <- vec2var(ca.jo_result_max_eig, r = 1)
vec2var_model

```



```

class(reduced_VAR)
class(vec2var_model)

# svar_model <- SVAR(vec2var_model, estmethod = "direct", Amat = Amat)
# irf_result <- irf(svar_model, n.ahead = 10, boot = TRUE, ci = 0.68)
# plot(irf_result)

# arch.test(reduced_VAR)
# serial.test(reduced_VAR)
# normality.test(reduced_VAR)

plot_irf_with_ci <- function(IRF_name) {

  # Extract response and confidence intervals
  irf_data <- as.data.frame(IRF_name$irf)
  irf_lower <- as.data.frame(IRF_name$Lower)
  irf_upper <- as.data.frame(IRF_name$Upper)

  # Create time index
  irf_data$Time <- seq_len(nrow(irf_data))
  irf_lower$Time <- irf_data$Time
  irf_upper$Time <- irf_data$Time

  # Reshape to long format
  irf_long <- pivot_longer(irf_data, cols = -Time, names_to = "Variable", values_to = "IRF")
  lower_long <- pivot_longer(irf_lower, cols = -Time, names_to = "Variable", values_to = "Lower")
  upper_long <- pivot_longer(irf_upper, cols = -Time, names_to = "Variable", values_to = "Upper")

  # Merge and label
  irf_combined <- irf_long %>%
    left_join(lower_long, by = c("Time", "Variable")) %>%
    left_join(upper_long, by = c("Time", "Variable")) %>%
    mutate(
      Shock = sub("\\\\.\\.*", "", Variable),
      Affected_Var = sub("\\.*\\.\\.", "", Variable)
    )

  # Plot IRFs by shock

```

```

shock_names <- unique(irf_combined$Shock)

for (shock in shock_names) {
  p <- irf_combined %>%
    filter(Shock == shock) %>%
    ggplot(aes(x = Time, y = IRF)) +
    geom_line(size = 1.2, color = "#1f77b4") +
    geom_line(aes(y = Lower), linetype = "dashed", color = "#7f7f7f", size = 0.9) +
    geom_line(aes(y = Upper), linetype = "dashed", color = "#7f7f7f", size = 0.9) +
    geom_hline(yintercept = 0, color = "black", size = 1.1) +
    facet_wrap(~ Affected_Var, scales = "free_y") +
    theme_minimal(base_family = "serif") +
    theme(
      plot.title = element_text(size = 16, face = "bold"),
      axis.title.x = element_text(size = 14),
      axis.title.y = element_text(size = 14),
      axis.text = element_text(size = 12),
      panel.grid.major = element_line(color = "gray85"),
      panel.grid.minor = element_blank(),
      strip.text = element_text(size = 14, face = "bold")
    ) +
    labs(
      title = paste("Impulse Response for Shock:", shock),
      x = "Time",
      y = "Response"
    ) +
    scale_y_continuous(expand = expansion(mult = c(0, 0.05)))

  print(p)
}
}

plot_irf_with_ci(irf_result)
# plot_irf_with_ci(irf_result2)

library(knitr)

# Create named vector of means
means <- c(
  mean(model_data$dif_log_expenditure, na.rm = TRUE),
  mean(model_data$dif_interest_rate, na.rm = TRUE),
  mean(model_data$dif_log_ERI, na.rm = TRUE),
  mean(model_data$dif_log_GDP, na.rm = TRUE),
  mean(model_data$dif_log_revenue, na.rm = TRUE),
  mean(model_data$dif_log_deflator, na.rm = TRUE)
)

```

```

)

# Variable names
variables <- c(
  "dif_log_expenditure",
  "dif_interest_rate",
  "dif_log_ERI",
  "dif_log_GDP",
  "dif_log_revenue",
  "dif_log_deflator"
)

# Assemble and display
mean_table <- data.frame(Variable = variables, Mean = round(means, 5))
kable(mean_table, align = "lr", caption = "Mean Values of Differenced Variables")

run_var_analysis <- function(model_data_type) {
  # Extract suffix from input name
  input_name <- deparse(substitute(model_data_type))
  suffix <- sub("^[_]+_", "", input_name)
  suffix <- ifelse(nchar(suffix) == 0, "data", suffix)

  # Split data into pre- and post-2008 samples
  clean_data_a <- model_data_type %>%
    filter(Year < 2008 & complete.cases(.))
  clean_data_b <- model_data_type %>%
    filter(!is.na(Year) & Year >= 2008)

  assign(paste0("clean_data_a_", suffix), clean_data_a, envir = .GlobalEnv)
  assign(paste0("clean_data_b_", suffix), clean_data_b, envir = .GlobalEnv)

  # Choose VAR specification
  var_a <- VAR(clean_data_a[, -c(1,2)], p = 1, type = "const")
  var_b <- VAR(clean_data_b[, -c(1,2)], p = 1, type = "const")

  assign(paste0("reduced_VAR_a_", suffix), var_a, envir = .GlobalEnv)
  assign(paste0("reduced_VAR_b_", suffix), var_b, envir = .GlobalEnv)

  # Check stability
  print(roots(var_a))
  print(roots(var_b))

  # Recursive identification matrix

```

```

k <- ncol(clean_data_a[,-c(1,2)])
Amat <- diag(1, k)
Amat[upper.tri(Amat)] <- NA

# SVAR estimation
svar_a <- SVAR(var_a, Amat = Amat, estmethod = "direct")
svar_b <- SVAR(var_b, Amat = Amat, estmethod = "direct")

assign(paste0("svar_model_a_", suffix), svar_a, envir = .GlobalEnv)
assign(paste0("svar_model_b_", suffix), svar_b, envir = .GlobalEnv)

# IRFs
irf_a <- irf(svar_a, n.ahead = 10, ci = 0.68, boot = 5000, cumulative = TRUE)
irf_b <- irf(svar_b, n.ahead = 10, ci = 0.68, boot = 5000, cumulative = TRUE)

assign(paste0("irf_result_a_", suffix), irf_a, envir = .GlobalEnv)
assign(paste0("irf_result_b_", suffix), irf_b, envir = .GlobalEnv)

# FEVDs
fevd_a <- fevd(svar_a, n.ahead = 10)
fevd_b <- fevd(svar_b, n.ahead = 10)

assign(paste0("FEVD_result_a_", suffix), fevd_a, envir = .GlobalEnv)
assign(paste0("FEVD_result_b_", suffix), fevd_b, envir = .GlobalEnv)

# Plot FEVDs
plot(fevd_a)
plot(fevd_b)
}

run_var_analysis(model_data)
# run_var_analysis(model_data2)

# run the chow test

# plot_irf_with_ci(irf_result2_data)

# Apply processing and plotting functions

```

References

- [1] O.J. Blanchard and D. Leigh. “Growth forecast errors and fiscal multipliers”. In: *American Economic Review* 103.3 (2013), pp. 117–120.
- [2] E. Ilzetzi, E.G. Mendoza, and C.A. Végh. “How big (small?) are fiscal multipliers?” In: *Journal of Monetary Economics* 60.2 (2013), pp. 239–254.
- [3] D. Sutherland, P. Hoeller, and R. Merola. “Fiscal consolidation: How much, how fast and by what means?” In: (2012).
- [4] T. Warmedinger, C.D. Checherita-Westphal, and P.H. De Cos. “Fiscal multipliers and beyond”. In: *ECB Occasional Paper* 162 (2015).