Empirical Analysis II Problem Set 4

Chase Abram, Jeanne Sorin*

July 5, 2020

1 Problem 1

1.1

. . . .

Represent y_t in terms of the z_t

$$z_t = y_t - y_{t-1}$$
$$y_t = \sum_{s=1}^{t} z_s$$

Calculate the variance $\sigma^2_{y,t}$ of y_t

$$\begin{aligned}
y_2 &= z_1 + z_2 & &\Rightarrow V(y_2) = V(z_1) + V(z_2) + 2cov(z_1, z_2) = 2\gamma_0 + 2\gamma_1 \\
y_3 &= z_1 + z_2 + z_3 & &\Rightarrow V(y_3) = V(z_1) + V(z_2) + V(z_3) + 2cov(z_1, z_2) + 2cov(z_1, z_3) + 2cov(z_2, z_3) \\
& & &\Rightarrow V(y_3) = 3\gamma_0 + 2^2\gamma_1 + 2^1\gamma_2 \\
&\vdots \\
\sigma^2_{y,t} &= t\gamma_0 + 2\sum_{s=1}^{t-1}(t-s)\gamma_s
\end{aligned}$$

So we represent a non-stationary process in terms of moments of a stationary process.

1.2

. . . .

$$\begin{aligned}
\nu_y^2 &= \lim_{t\to\infty} \frac{\sigma^2_{y,t}}{t} \\
&= \lim_{t\to\infty} \frac{t\gamma_0}{t} + \frac{2}{t}\sum_{s=1}^{t-1}(t-s)\gamma_s \\
&= \lim_{t\to\infty} \gamma_0 + 2\sum_{s=1}^{t-1}\frac{t-s}{t}\gamma_s
\end{aligned}$$

*This solution set benefited greatly from the comments and points raised by Manav Chaudhary, George Votja, and Tom Hierons.

1.3

. . . .

The idea here is that we look at the convergence of the process $\frac{\sigma^2_{y,t}}{t}$.

$$\begin{aligned}
\nu_y^2 &= \lim_{t\to\infty} \frac{\sigma^2_{y,t}}{t} \\
&= \lim_{t\to\infty} \sigma^2_{y,t} \cdot \lim_{t\to\infty} \frac{1}{t} && \text{(Since both sequences converge)} \\
&= \sigma^2_y \cdot \lim_{t\to\infty} \frac{1}{t} \\
&= \sigma^2_y \cdot 0 \\
&= 0
\end{aligned}$$

Where we know that $\sigma_y^2 < \infty$, since it is given in the question. So we have the result $\nu_y^2 = 0$.

Perhaps more importantly, we know

$$0 = \gamma_0 + \lim_{t\to\infty} 2\sum_{s=1}^{t-1} \frac{(t-s)}{t}\gamma_s$$

The second term must therefore be a sequence which converges to $-\gamma_0$. Note that it would not technically be correct to call it a series, as each time we increase t, we increase the number of terms *but also* change the nature of every term. Nonetheless, we may conclude

$$\lim_{t\to\infty} 2\sum_{s=1}^{t-1} \frac{(t-s)}{t}\gamma_s = 2\sum_{s=1}^{\infty}\gamma_s$$

because the left sum converges term-by-term to the terms in the right series. If the right side were infinite (positive WLOG) or non-convergent, so also would be the left side, because then for any t we could simply pick t' large enough so that $\sum_{s=1}^{t'} \frac{t'-s}{t'}\gamma_s$ is sufficiently close to $\sum_{s=1}^{t}\gamma_s$. But we know the left side is finite and convergent, from the question's assumption, so the right side must also be finite and converge. Then we have the result

$$\sum_{s=-\infty}^{\infty}\gamma_s = 0$$

1.4

....

We have that

$$\begin{aligned} S_z(\omega) &= \frac{1}{2\pi}\sum_{j=-\infty}^{\infty}\gamma_j e^{-ij\omega} \\ \implies S_z(0) &= \frac{1}{2\pi}\sum_{j=-\infty}^{\infty}\gamma_j \\ &= 0 \end{aligned}$$

Getting $S_z(0) = 0$ would make sense in this setting: by taking the first difference, we artificially *kill* all variance associated with cycles of "length ∞" (the trend), which is usually captured by the spectrum at frequency 0.

1.5

....

$$\begin{aligned} y_t &= \rho y_{t-1} + \varepsilon_t \\ &= \frac{1}{1-\rho L}\varepsilon_t \text{ we can invert the lag because } |\rho| < 1 \end{aligned}$$

Moreover

$$\begin{aligned} z_t &= y_t - y_{t-1} \\ &= \frac{1}{1-\rho L}\varepsilon_t - \frac{1}{1-\rho L}\varepsilon_{t-1} \\ &= \frac{1}{1-\rho L}\varepsilon_t - \frac{L}{1-\rho L}\varepsilon_t \\ &= \frac{1-L}{1-\rho L}\varepsilon_t \\ &= \left[1 + \sum_{j=1}^{\infty}\rho^j(1-\rho)L^j\right]\epsilon_t \end{aligned}$$

Which is the Wold decomposition, so $\theta(L) = 1 + \sum_{j=1}^{\infty}\rho^j(1-\rho)L^j$. To check that this is the true Wold representation, and not just some MA representation, we may note that $c_0 = 1$, and the roots of the process are invertible (implied by $|\rho| < 1$, thus $|\rho^j(1-\rho)| < 1$).
We can also express it as a ARMA:

$$\begin{aligned} z_t &= y_t - y_{t-1} \\ z_t &= \frac{1}{1-\rho L}\varepsilon_t - \frac{1}{1-\rho L}e_{t-1} \\ z_t &= \frac{1}{1-\rho L}(\varepsilon_t - \varepsilon_{t-1}) \\ (1-\rho L)z_t &= \varepsilon_t - \varepsilon_{t-1} \\ &= \varepsilon_t + \xi\varepsilon_{t-1} \end{aligned}$$

With $p(L) = (1-\rho L)$ and $\xi = (-1)$.

1.6

....

We calculate the spectrum s_z of z_t. The spectrum is the Fourier Transform of the series $\{\gamma_j\}$, i.e. the Fourier transform of the autocovariances divided by 2π.

$$\begin{aligned} s_z(\omega) &= \frac{1}{2\pi}\sum_{j=-\infty}^{\infty}\gamma_j e^{-i\omega j} \\ &= \frac{\gamma_0}{2\pi} + \sum_{j=1}^{\infty}\gamma_j(e^{-ij\omega} + e^{ij\omega}) \\ &= \frac{\gamma_0}{2\pi} + \sum_{j=1}^{\infty}2\gamma_j\cos(j\omega) \end{aligned}$$

Note that the spectrum of an ARMA process $\phi(L)x_t = \theta(L)y_t$ is

$$s_x(w) = s_y(w)\frac{|\theta(e^{-iw})|^2}{|\phi(e^{-iw})|^2}$$

In our case

$$\begin{aligned} s_z(w) &= \frac{\sigma^2}{2\pi}\frac{|(1-e^{-iw})|^2}{|(1-\rho e^{-iw})|^2} \\ &= \frac{\sigma^2}{2\pi}\frac{(1-e^{-iw})(1-e^{iw})}{(1-\rho e^{-iw})(1-\rho e^{iw})} \\ &= \frac{\sigma^2}{2\pi}\frac{2-2\cos(w)}{1-2\rho\cos(w)+\rho^2} \end{aligned}$$

1.7

....

Calculate the value of the spectrum $s_z(0)$ at frequency 0:

$$\begin{aligned} s_z(0) &= \frac{\sigma^2}{2\pi}\frac{2-2\cos(0)}{1-2\rho\cos(0)+\rho^2} \\ &= \frac{\sigma^2}{2\pi}\frac{2-2}{1-2\rho+\rho^2} = 0 \end{aligned}$$

The quick and easy relationship we can notice is that this result agrees with our more general result, because we above considered a the difference of a general stationary process, and the AR(1), y_t, here is stationary.

Let's try to understand what is at play here. First, let's rewrite the spectrum of z_t to reveal the spectrum of y_t

$$s_z(w) = |\theta(e^{-iw})|^2 s_y(w)$$
$$\text{Where } s_y(w) = |\phi^{-1}(e^{-iw})|^2 s_\epsilon(w)$$

And the spectrum of y_t, $s_y(w)$ is well defined because ϵ_t is a stationary process.
A common interpretation for the first difference $z_t = y_t - y_{t-1}$ is a *filter*. Filters extract parts of the

process y_t that are associated with specific frequency bands. Here, the first difference filter (Δ^1) corresponds to removing the role of frequency bands corresponding to movements that are common to all periods, i.e. long run movements (which one can think of as period ∞), which correspond to $\omega = 0$.

The gain of the filter at frequency 1 is equal to 0 because *once filtered, the data is not informative about variations of period* ∞.

In other words, the first-difference z_t wants to extract all variation corresponding to single-period (from t to $t+1$) periods. His spectrum at frequency 0 is equal to 0 because for low frequencies, i.e. high periods, no information is left after applying the filter. Therefore one should not put any weight on these frequencies when analysing period-to-period variations. The first difference filter removes indiscriminately information from any frequency. What remains is only idiosyncratic shocks.

1.8

. . . .

$$\theta(L) = \frac{1-L}{1-\rho L}$$
$$\theta(1) = 0$$

Alternatively

$$\theta(L) = \sum_{j=0}^{\infty} \rho^j L^j (1-L)$$
$$\Rightarrow \theta(1) = 0$$

where we must note that we are using the assumption $|\rho| < 1$ to write $\theta(L)$ as a series.

1.9

. . . .

Redo this part, assuming that $\rho = 1$. How do your results differ ?

When $\rho = 1$, we find $z_t = y_t - y_{t-1} = \epsilon_t$. Therefore we have $p(L) = 1$, $\xi = 0$, $\theta(L) = 1$, using the notation from part 5. The spectrum is the simply $s_z(\omega) = \frac{\sigma^2}{2\pi}$, and this result applies to $\omega = 0$. Since $\theta(L) = 1$, we have the specific result $\theta(1) = 1$[1]

$$z_t = \frac{1-L}{1-L}\varepsilon_t = \varepsilon_t$$
$$\theta(L) = \theta(1) = 1$$

Since we have $y_t = y_{t-1} + \epsilon_t$ is a random walk, first differencing just leaves the white noise shock process, which is stationary i.e. the unit root process is an $I(1)$ autoregressive process and first

[1] Technically, θ is mapping an operator to a polynomial in that operator, so the "1" actually denotes the identity operator i.e. $1x_t = x_t$.

differencing gives us a $I(0)$ process. This result can also be seen in the spectrum of z we calculated. If $\rho = 1$, then we don't gain anything by looking at first difference rather than by looking directly at y_t because, by definition, white noise process makes periods totally independent from each other.

1.10

....

Our old process

$$z_t = \frac{1-L}{1}\varepsilon_t$$

Calculate the autocovariances γ_j for z_t [2].
Using the formula for covariances of an ARMA process following $\phi(L)x_t = \theta(L)y_t$, with $\theta = 0$ and $\phi = (-1)$ we have

$$\begin{aligned}\gamma_0 &= \frac{1+2\theta\phi+\theta^2}{1-\phi^2}var(y_t)\\ &= \frac{1+1}{1-\rho^2}\sigma^2\\ \gamma_0 &= 2\sigma^2\end{aligned}$$

and

$$\begin{aligned}\gamma_1 &= \frac{(\rho-1)(1-\rho)}{(1-\rho^2)}\sigma^2\\ \gamma_1 &= -\sigma^2\\ \gamma_s &= 0 && \forall s \geq 2\end{aligned}$$

Regarding our new process:

$$\begin{aligned}\tilde{z_t} &= \varepsilon_t - \alpha e_{t-1}\\ &= (1-\alpha L)e_t\\ \tilde{y_0} &= 0, \tilde{y_t} = \tilde{y_{t-1}} - \tilde{z}_t\end{aligned}$$

Therefore we have

$$\begin{aligned}\tilde{\gamma_0} &= Var(\varepsilon_t - \alpha\varepsilon_{t-1})\\ &= (1+\alpha^2)\sigma^2 \text{ because the errors are uncorrelated}\\ \tilde{\gamma_1} &= -\alpha\sigma^2\\ \tilde{\gamma_s} &= 0 && \forall s \geq 2\end{aligned}$$

1.11

....

Spectral density at frequency zero for z_t and $\tilde{z}_t$. From question 7 we already have that

$$s_z(0) = 0$$

[2]We recognize that we could do this directly, but our approach gives more practice with the ARMA representation.

but we can easily check

$$
\begin{aligned}
s_z(0) &= \frac{1}{2\pi}\sum_{j=-\infty}^{\infty}\gamma_j \\
&= \frac{1}{2\pi}(-\sigma^2 + 2\sigma^2 - \sigma^2) \\
&= 0
\end{aligned}
$$

Moreover, $\tilde{z}_t$ is simply a MA(1) process so the spectrum of $\tilde{z}_t$ is

$$S_{\tilde{z}}(\omega) = (1 - 2\alpha\cos(\omega) + \alpha^2)\frac{\sigma^2}{2\pi}$$

Therefore

$$S_{\tilde{z}}(0) = (1-\alpha)^2\frac{\sigma^2}{2\pi}$$

Again, we can easily check

$$
\begin{aligned}
s_{\tilde{z}}(0) &= \frac{1}{2\pi}\sum_{j=-\infty}^{\infty}\gamma_j \\
&= \frac{1}{2\pi}(-\alpha\sigma^2 + (1+\alpha^2)\sigma^2 - \alpha\sigma^2) \\
&= (1-\alpha)^2\frac{\sigma^2}{2\pi}
\end{aligned}
$$

The spectrum at frequency 0 is $\neq 0$: we are looking at a white noise process with variance lower than the white noise process ϵ_t!

1.12

. . . .

$$
\begin{aligned}
z_t &= \theta(L)\varepsilon_t \\
\tilde{z}_t &= \theta(\tilde{L})\varepsilon_t
\end{aligned}
$$

With $\theta(L) = (1-L)$ and $\theta(\tilde{L}) = (1-\alpha L)$.

$$
\begin{aligned}
\theta(1) &= 0 \\
\theta(\tilde{1}) &= 1-\alpha
\end{aligned}
$$

1.13

....

Calculate the LR variances.

Note that $y_t = \epsilon_t$, so y_t is stationary, and we have

$$\begin{aligned} asy\,\text{var}(y_t) &= \lim_{t\to\infty} \frac{\text{var}(\epsilon_t)}{t} \\ &= 0 \end{aligned}$$

On the other hand $\tilde{y}_t$ is not stationary, so we must consider the asymptotics. We may notice, however, that we are exactly in the situation described above in parts 1 - 4, so we may use our formula from there.

$$\begin{aligned} asy\,\text{var}(\tilde{y}_t) &= \lim_{t\to\infty} \frac{\text{var}(\tilde{y}_t)}{t} \\ &= \lim_{t\to\infty} \tilde{\gamma}_0 + 2\sum_{j=1}^{t-1} \frac{t-j}{t}\tilde{\gamma}_j \\ &= \tilde{\gamma}_0 + \lim_{t\to\infty} 2(\frac{t-1}{t}\tilde{\gamma}_1) \\ &= \sigma^2(1+\alpha^2) + \lim_{t\to\infty} 2(\frac{t-1}{t}(-\alpha\sigma^2)) \\ &= \sigma^2(1+\alpha^2) + 2(-\alpha\sigma^2) \\ &= \sigma^2(1-\alpha^2) \end{aligned}$$

We may note that, as $\alpha \to 1$, we have $asy\,\text{var}(\tilde{y}_t) \to asy\,\text{var}(y_t)$

1.14

Given a finite sample, could you empirically distinguish y_t from $\tilde{y}_t$, when α is very close to 1?

When $\alpha \to 1$, $LR(Var(\tilde{y}_t)) \to 0 = LR(Var(y_t))$ so we could have troubles distinguishing empirically y_t from $\tilde{y}_t$. In other words, due to idiosyncratic draws, we might not be able to tell the difference empirically between the two DGPs, and this issue is aggravated the closer α is to 1.

1.15

Given these results, comment slide 100.

Slide 100 tells us that :
Consider a test of H_0: "y is I(0)" against a nonempty alternative of I(1) processes. Suppose the test has power above $1-\beta$, i.e. if truth is an element $\tilde{y}$ of the alternative, the null hypothesis is rejected with probability above $1-\beta$? Find y, which is I(0) and close to $\tilde{y}$ in the L_1-distance. If it is close enough, then the test will reject the null hypothesis with probability above $1-\beta$, if y is true. Therefore, to obtain power of, say, 95%, when size is 5%, the null hypothesis needs to be restricted further.

Our results are consistent with what is above: to increase the power of the test we would need to restrict the set of elements in the null. Basically, we have shown that we can have I(0) processes which are, in some sense, really close to I(1) processes, and if these processes are close enough, we will reject them quite often in favor of an I(1) alternative hypothesis. So we really need to be sharper about cutting off our null hypothesis in the space of I(0) processes, otherwise we may technically have an I(0) process, but almost always reject the null that it is I(0). I believe this issue is due to the nature of the function spaces of I(0) and I(1) processes, i.e. for any I(1) process, there is an I(0) process arbitrarily close under the L_1 norm.

2 Problem 2

2.1

. . . .

$$\begin{aligned} y_t &= \rho y_{t-1} + \varepsilon_t - \theta\varepsilon_{t-1} \\ (1-\rho L)y_t &= (1-\theta L)\varepsilon_t \\ y_t &= \frac{1-\theta L}{1-\rho L}\varepsilon_t \end{aligned}$$

Where we can invert $1-\rho L$ because we know that $|\rho| < 1$.

2.2

. . . .

$$\begin{aligned} (1-\rho L)y_t &= (1-\theta L)\varepsilon_t \\ |(1-\rho e^{-iw})|^2 S_y(w) &= |(1-\theta e^{-iw})|^2 S_\varepsilon(w) \\ (1-\rho e^{-iw})(1-\rho e^{iw})S_y(w) &= (1-\theta e^{-iw})(1-\theta e^{iw})S_\varepsilon(w) \\ (1-2\rho\cos(w)+\rho^2)S_y(w) &= (1-2\theta\cos(w)+\theta^2)S_\varepsilon(w) \end{aligned}$$

Because ε_t is white noise we know that $S_\varepsilon(w) = \frac{\sigma^2}{2\pi} = \frac{1}{2\pi}$.

$$S_y(w) = \frac{1}{2\pi}\frac{1-2\theta\cos(w)+\theta^2}{1-2\rho\cos(w)+\rho^2}$$

2.3

. . . .

On the red curve ($\rho = 0.9, \theta = 0.91$), we observe that the closer to 0, the lower, the spectrum is. While moving away from 0, the spectrum converges to $\frac{1}{2\pi}$ and becomes flatter. What happens is that close to 0, the strong persistence of the shocks ε_t, $\theta = 0.91$ is "stronger" than the persistence of the y_t, $\rho = 0.9$, and therefore drives down the importance of very short frequencies (small $\omega \Leftrightarrow$ long cycles).
This suggests than in such a process, we can discard long cycles because idiosyncratic shocks (the moving average part) matter more than persistence of y_t (the auto-regressive part).

The logic goes the other way around for the blue line ($\rho = 0.9, \theta = 0.89$).

2.4

. . . .

If $\rho = \theta$ then we are back to the white noise process case: a flat spectrum. This means that we put similar weight on each frequency: no cycle is "stronger" than another because the negative persistence of the shocks and of the y_t exactly "cancel each other".

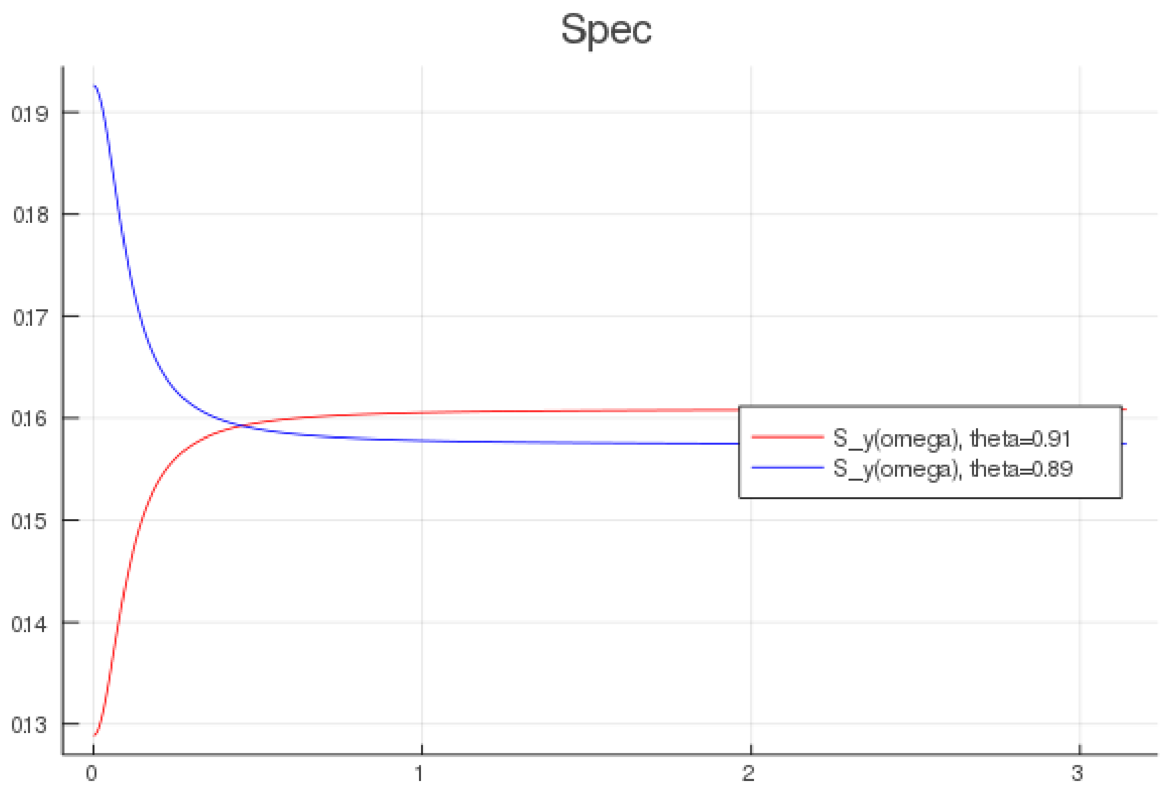


Figure 1: Spectra of this process

3 Problem 3

3.1

....

We can start by recalling that for a stationary AR(1) process (obviously $|0.9| < 1$) with $\text{var}(\epsilon) = \sigma^2$, the spectrum is fairly easy to calculate. The reason is that the covariances γ_j are simply $\frac{\rho^j}{1-\rho^2}$, and $e^{i\omega} = \cos(\omega) + i\sin(\omega)$, so with a few minutes of algebra we find

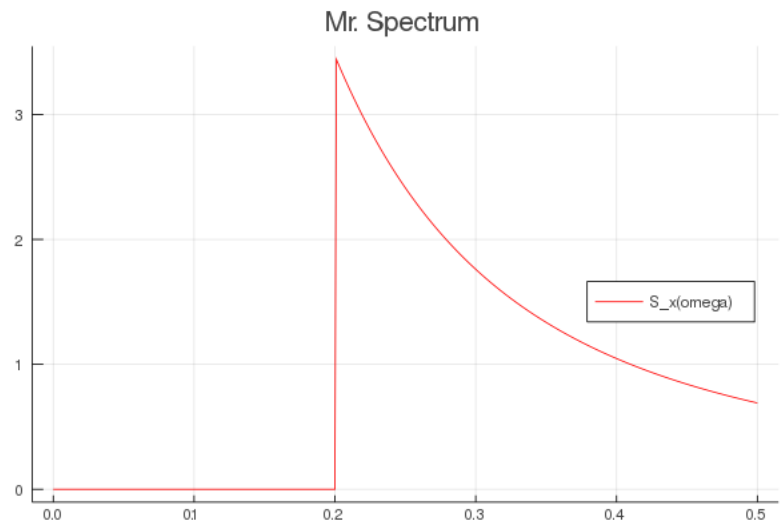
$$
\begin{aligned}
s_x(\omega) &= \frac{1}{2\pi}\sum_{j=-\infty}^{\infty}\gamma_j e^{-i\omega j} \\
&= \frac{\sigma^2}{2\pi}\frac{1}{1-2\rho\cos(\omega)+\rho^2}
\end{aligned}
$$

We are given $y_t = h(L)x_t$. Therefore

$$
\begin{aligned}
s_y(\omega) &= s_{h(L)x}(\omega) \\
&= 2\pi\tilde{h}(\omega)s_x(\omega) \\
&= \frac{\sigma^2}{2\pi}\frac{1}{1-2\rho\cos(\omega)+\rho^2}\mathbf{1}_{|\omega|\geq\underline{\omega}}
\end{aligned}
$$

With our given parameters, the spectrum becomes

$$s_y(\omega) = \frac{1}{2\pi} \frac{1}{1 - 1.8\cos(\omega) + 0.81} \mathbf{1}_{|\omega| \geq 0.2}$$



3.2

. . . .

The unconditional variance of $x_t = \frac{1}{1-0.9^2} \approx 5.25$. To find the unconditional variance of y_t, we may note that

$$\begin{aligned}\gamma_0^y &= \int_{-\pi}^{\pi} s_y(\omega) \mathrm{d}\omega \\ &= 2\int_{\underline{\omega}}^{\pi} s_x(\omega) \mathrm{d}\omega \\ &\approx 1.62\end{aligned}$$

Therefore the unconditional variance of y_t is about $\frac{1.62}{5.25} \approx 31\%$ of the unconditional variance of x_t. It is unsurprising that the filter kills so much of the variance, as $\rho = 0.9$ implies that most of the variance in x_t is coming from low-frequency movement, and our filter annihilates a lot of this "trend" variance.

3.3

. . . .

Consider $j = 0$. Then

$$\begin{aligned}
h_0 &= \int_{-\pi}^{\pi} \tilde{h}(\omega)\mathrm{d}\omega \\
&= \frac{1}{2\pi}\int_{-\pi}^{\pi} \mathbf{1}_{|\omega|>\underline{\omega}}\mathrm{d}\omega \\
&= \frac{1}{\pi}\int_{\underline{\omega}}^{\pi} 1\mathrm{d}\omega \\
&= \frac{\pi-\underline{\omega}}{\pi}
\end{aligned}$$

Now consider $j \neq 0$. Then

$$\begin{aligned}
h_j &= \int_{-\pi}^{\pi} \tilde{h}(\omega)e^{i\omega j}\mathrm{d}\omega \\
&= \frac{1}{2\pi}\Big[\int_{-\pi}^{-\underline{\omega}} e^{i\omega j}\mathrm{d}\omega + \int_{\underline{\omega}}^{\pi} e^{i\omega j}\mathrm{d}\omega\Big] \\
&= \frac{1}{2\pi}\Big[\int_{-\pi}^{-\underline{\omega}} \cos(\omega j)\mathrm{d}\omega + i\int_{-\pi}^{-\underline{\omega}} \sin(\omega j)\mathrm{d}\omega + \int_{\underline{\omega}}^{\pi} \cos(\omega j)\mathrm{d}\omega + i\int_{\underline{\omega}}^{\pi} \sin(\omega j)\mathrm{d}\omega\Big] \\
&= \frac{1}{2\pi}\Big[\int_{-\pi}^{-\underline{\omega}} \cos(\omega j)\mathrm{d}\omega + \int_{\underline{\omega}}^{\pi} \cos(\omega j)\mathrm{d}\omega\Big] \\
&= -\frac{\sin(\underline{\omega} j)}{j\pi}
\end{aligned}$$

Moving to the last line just requires some tedious calculus, as well as knowledge about the periodicity of sine.

So we have defined h_j for all $j \in \mathbb{Z}$.

3.4

....

We can use the h_j we just computed, as well as the symmetry $h_j = h_{-j}$.

$$\begin{aligned}
\tilde{h}(\omega) &= \frac{1}{2\pi}\sum_{j=-\infty}^{\infty} h_j e^{-i\omega j} \\
&= \frac{1}{2\pi}\Big[\frac{\pi-\underline{\omega}}{\pi} + \sum_{j=1}^{\infty} h_j(e^{-i\omega j}+e^{i\omega j})\Big] \\
&= \frac{1}{2\pi}\Big[\frac{\pi-\underline{\omega}}{\pi} + \sum_{j=1}^{\infty} -\frac{\sin(\underline{\omega} j)}{j\pi}2\cos(\omega j)\Big] \\
&= \sum_{j=0}^{\infty} \kappa_j \cos(\omega j)
\end{aligned}$$

where

$$\kappa_0 = \frac{1}{2\pi}\left[\frac{\pi - \underline{\omega}}{\pi}\right]$$
$$\kappa_j = \frac{1}{2\pi}\left[-\frac{2\sin(\underline{\omega} j)}{j\pi}\right] \qquad (j \neq 0)$$

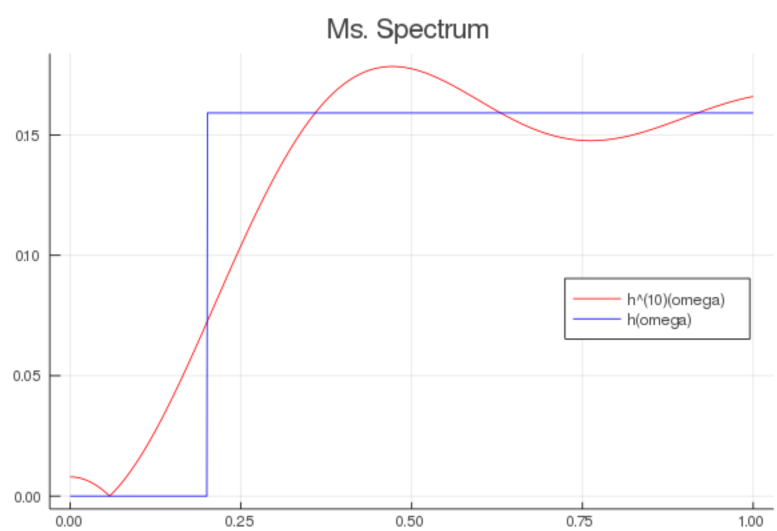
3.5

....

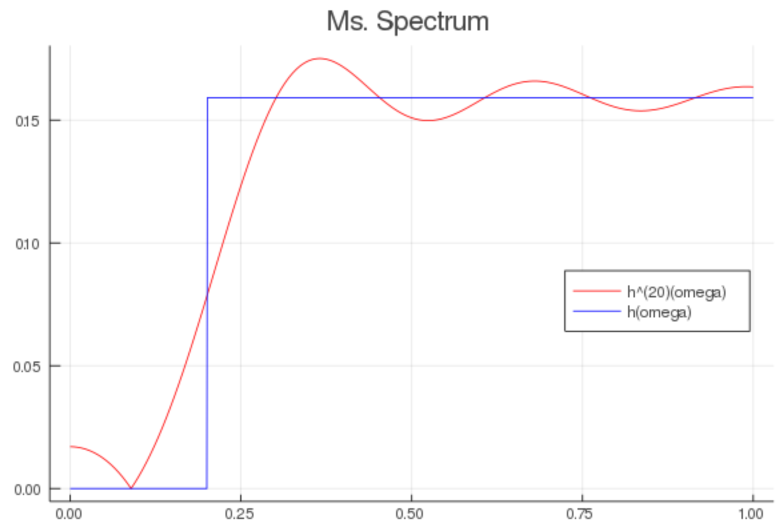
The finite approximation of $\tilde{h}$ is fairly easy to code up, given that we know the κ_j. The obvious first observation is that any finite approximation will have to be "off" in the sense that the jump from 0 to $\frac{1}{2\pi}$ must be a smooth curve, but that adding more lags/leads improves the approximations[3].

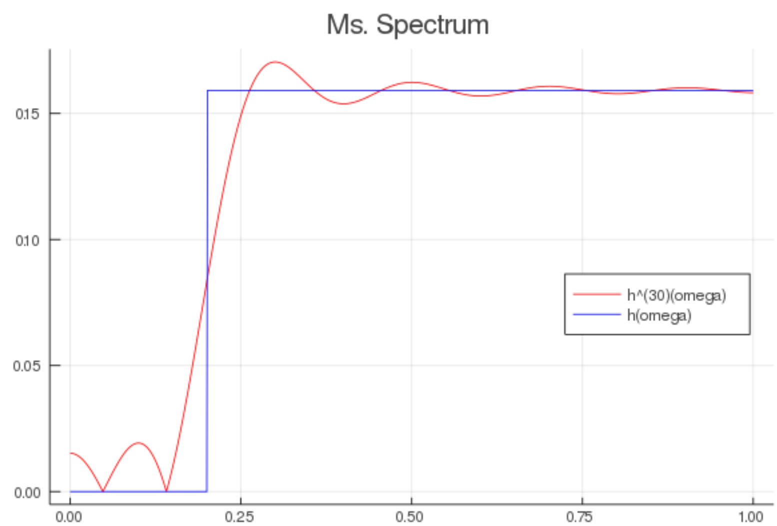
We can also see that the approximation is worse, the closer we are to the jump point. This detail is likely quite important for determining how useful a bandpass filter will be in helping to filter out low-frequencies. For example, if we know that there is some super low frequency we wish to filter, and it is nowhere near the frequencies we care about, then a finite filter like the ones above will likely be at least somewhat useful, but mainly it will give the same result as if we used the true bandpass filter. The issue would be if the frequencies we care about and the frequencies we don't care about are quite close, in which case the finite filter's deviation from the true filter might give it an edge, in that instead of fully truncating, the finite filter smoothly declines to zero.

The other limitation we need to think about here is how frequent our available data is. My impression is that quarterly is generally as good as it gets for macro, with the occasion that sometimes monthly data is available. Sometimes only annual data is available. In any case, we will be limited in how much we can cut low frequencies. So if we were looking for, say quarterly data indicating a recession, but wanted to cut out all frequencies lower than semi-annual, we might be playing a bit of a dangerous game in trying to filter so close to the frequency we care about, and in this case the finite filter might provide a happy medium over needing to know the exact frequency where we want to stop cutting, since the finite filter gives some wiggle room by smoothing around $\underline{\omega}$.



[3] We could formalize this by considering the sup norm $\|\cdot\|_\infty$, for example.


Ms. Spectrum
0.15
0.10
0.05
0.00
0.00
0.25
0.50
0.75
1.00
h^(20)(omega)
h(omega)


Ms. Spectrum
0.15
0.10
0.05
0.00
0.00
0.25
0.50
0.75
1.00
h^(30)(omega)
h(omega)

4 Problem 4

4.1

....

Done. All `Julia` code for this problem is included at the end of the problem.

4.2

....

I downloaded the `.csv` file, cleaned it, and reformatted the dates in a spreadsheets software. I then read the code into my `Jupyter` notebook workspace.

4.3

....

We find[4]

$$\hat{\mu} = \begin{bmatrix} -0.00011806241936085371 \\ 0.06830905156283507 \\ 0.044435601586584994 \end{bmatrix}$$

$$\hat{B} = \begin{bmatrix} 0.9651294456472783 & 0.012752109654163846 & 0.020746706879577914 \\ 0.10561390929538561 & 0.9696857314594531 & -0.07753790871309718 \\ 0.03210397273699073 & 0.0020975860492171705 & 0.9607191574778255 \end{bmatrix}$$

$$\hat{\Sigma} = \begin{bmatrix} 4.84666638372213e-5 & 3.667454130274951e-5 & 2.9406687773382987e-5 \\ 3.667454130274951e-5 & 0.00032399743279751714 & 2.614794131529033e-5 \\ 2.9406687773382984e-5 & 2.6147941315290335e-5 & 7.691857125861394e-5 \end{bmatrix}$$

4.4

....

More computed stuff

$$V = \begin{bmatrix} -0.3249926276392799 & 0.4383234202180775 & 0.5519823432597906 \\ 0.9259901548611698 & -0.5944736009024241 & 0.670123987693673 \\ 0.19215104756493886 & 0.674146658464189 & 0.4962351598254208 \end{bmatrix}$$

$$\hat{D} = \begin{bmatrix} 0.9165288679399272 & 0.0 & 0.0 \\ 0.0 & 0.9797431538735092 & 0.0 \\ 0.0 & 0.0 & 0.9992623127711215 \end{bmatrix}$$

$$\Rightarrow D = \begin{bmatrix} 0.9165288679399272 & 0.0 & 0.0 \\ 0.0 & 1.0 & 0.0 \\ 0.0 & 0.0 & 1.0 \end{bmatrix}$$

[4]You may find yourself wondering: why did he not just truncate the computed values to make them more readable? Well, the answer is

4.5

....

I include some other variables, for completeness. Note that normalizing β by its first component requires inverse scaling of α in order to maintain $\alpha\beta' = I - B$.

ν: [-0.3249926276392799, 0.9259901548611698, 0.19215104756493886]
ν^*: [0.4383234202180775 0.5519823432597906; -0.5944736009024241 0.670123987693673; 0.674146658464189 0.4962351598254208]
α: [0.04008674041224827, -0.11421775082054456, -0.023701181222568434]

$$\beta = \begin{bmatrix} 1.0 \\ -0.20703498581110913 \\ -0.832756858908041 \end{bmatrix}$$

$$\beta^* = \begin{bmatrix} -0.6544874706235589 & 1.461333754991795 \\ -0.5290153026795443 & 0.6002142663248722 \\ 1.4424035816070968 & -0.4208658009213407 \end{bmatrix}$$

$$\nu = \begin{bmatrix} -0.3249926276392799 \\ 0.9259901548611698 \\ 0.19215104756493886 \end{bmatrix}$$

$$\nu^* = \begin{bmatrix} 0.4383234202180775 & 0.5519823432597906 \\ -0.5944736009024241 & 0.670123987693673 \\ 0.674146658464189 & 0.4962351598254208 \end{bmatrix}$$

$$\alpha\beta' = \begin{bmatrix} 0.04008674041224827 & -0.008299357732463436 & -0.0333825080295659 \\ -0.11421775082054456 & 0.02364707042050824 & 0.09511561540485801 \\ -0.023701181222568434 & 0.004906973718120982 & 0.01973732122731633 \end{bmatrix}$$
$$= I - B$$

The *cay* relationship is

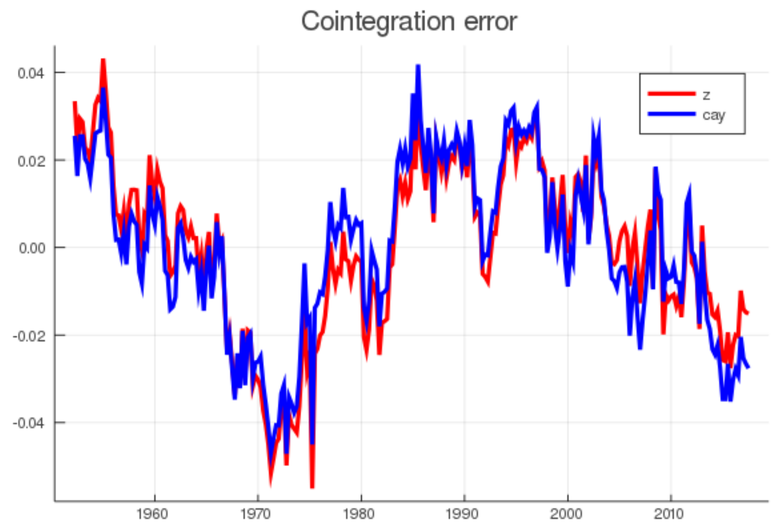
$$cay = c - 0.274092a - 0.757866y - 0.668337$$

It's far from a perfect match, but our β coefficients line up reasonably well with the *cay* coefficients. Part of this discrepancy may be due to the fact that that they find their *cay* equation by first testing for cointegration, then simply running an OLS regression as we do below.

4.6

....

Their *cay* is demeaned, so we also demean our z.



4.7

....

Our regression yields

$$
\begin{aligned}
\kappa &= -0.6093359825683429 \\
\beta_a &= 0.2463435266311229 \\
\beta_y &= 0.7851509539548708 \\
c &= \kappa + \beta_a a + \beta_y t + \epsilon
\end{aligned}
$$

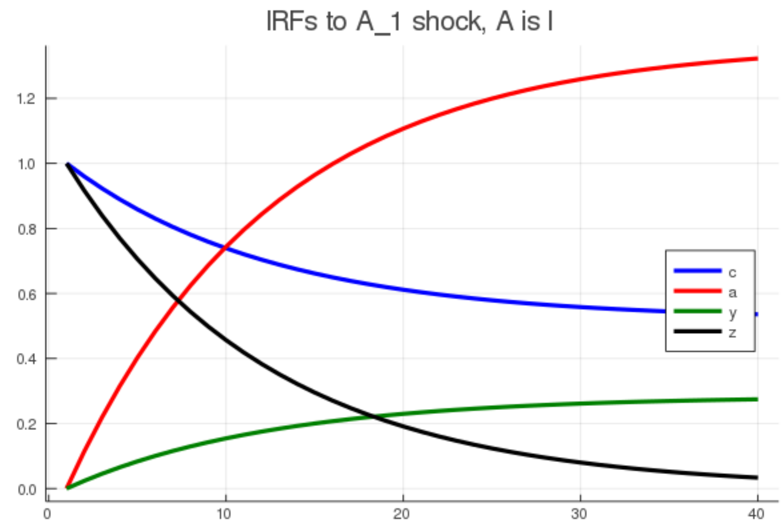
We can compare to the *cay* equation above, and again we find that our estimate is close (closer than before), but not exact[5].

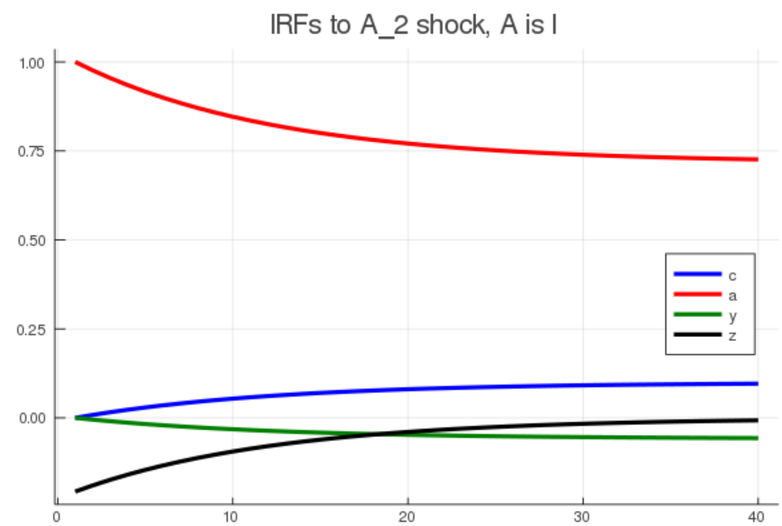
4.8

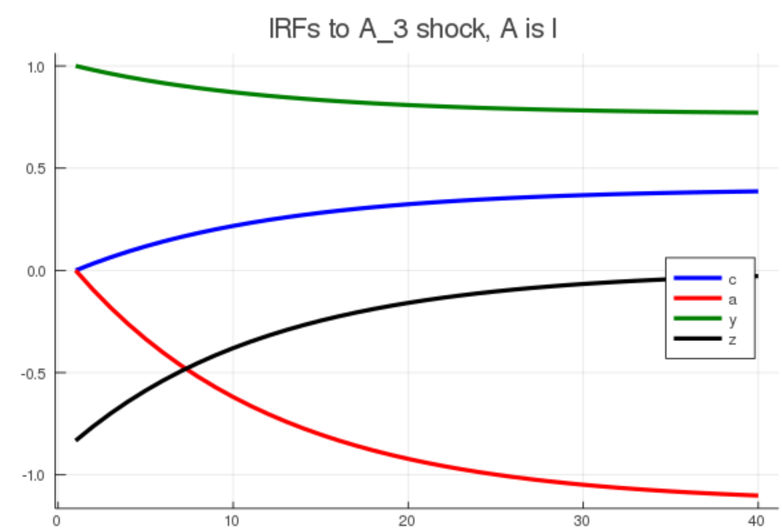
....

I assume Uhlig meant we should put each set of 4 IRFs together, not to put all 12 on one plot. Either way, the scale is included, so comparisons are easy.

[5]I actually worked some with this data while at the Fed Board, and was never able to find how they got the exact coefficients listed. My suspicion is that either they are doing more to the data then the original paper reveals, or there is some strange lag between when the data and estimate are updated. This paper made big waves, but it has also been picked to pieces.


IRFs to A_1 shock, A is I
c
a
y
z


IRFs to A_2 shock, A is I
c
a
y
z


IRFs to A_3 shock, A is I
c
a
y
z

4.9

....

It would be hard to argue that any variable moves more than assets, as it has the greatest movement in all three plots. It's not entirely clear how this makes it easier to forecast assets, but I suppose in some sense we have the least reservations about the direction of change with assets. For consumption and income, especially with the asset shock, the deviation from zero is not nearly as large as it is for assets.

4.10

....

Let's start with the algebra. We can recursively solve.

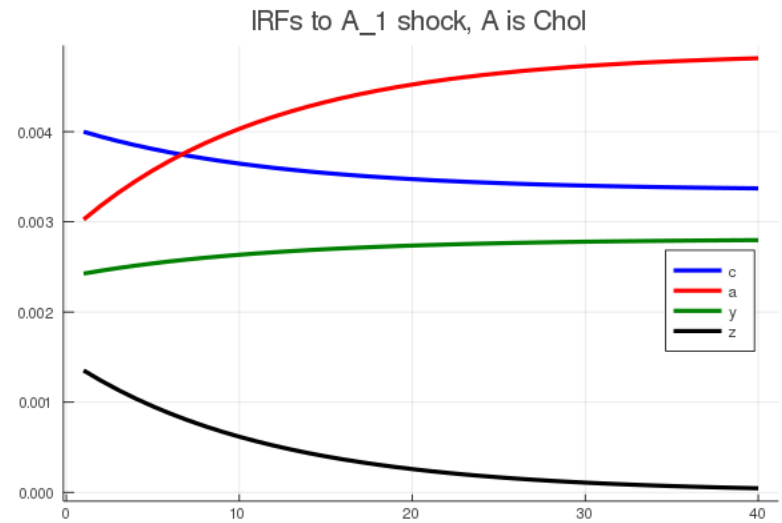
$$
\begin{aligned}
AA' &= \Sigma \\
\begin{bmatrix} A_{11} & 0 & 0 \\ A_{21} & A_{22} & 0 \\ A_{31} & A_{32} & A_{33} \end{bmatrix}
\begin{bmatrix} A_{11} & A_{21} & A_{31} \\ 0 & A_{22} & A_{32} \\ 0 & 0 & A_{33} \end{bmatrix}
&= \begin{bmatrix} \Sigma_{11} & \Sigma_{21} & \Sigma_{31} \\ \Sigma_{21} & \Sigma_{22} & \Sigma_{32} \\ \Sigma_{31} & \Sigma_{32} & \Sigma_{33} \end{bmatrix} \\
\Rightarrow A_{11} &= +\sqrt{\Sigma_{11}} && \text{(Sign of diag)} \\
A_{21} &= \Sigma_{21}/A_{11} \\
A_{22} &= +\sqrt{\Sigma_{22} - A_{21}^2} && \text{(Sign of diag)} \\
A_{31} &= \Sigma_{31}/A_{11} \\
A_{32} &= (\Sigma_{32} - A_{31}A_{21})/A_{22} \\
A_{33} &= +\sqrt{\Sigma_{33} - A_{31}^2 - A_{32}^2} && \text{(Sign of diag)}
\end{aligned}
$$

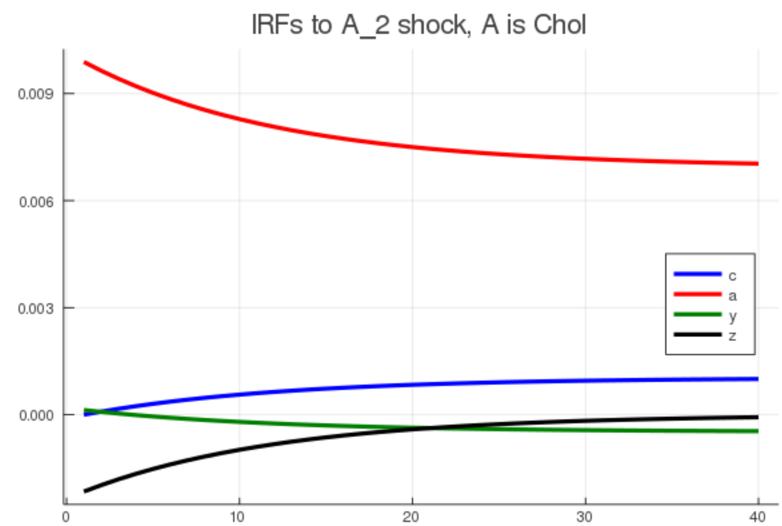
Here is the estimated A we find

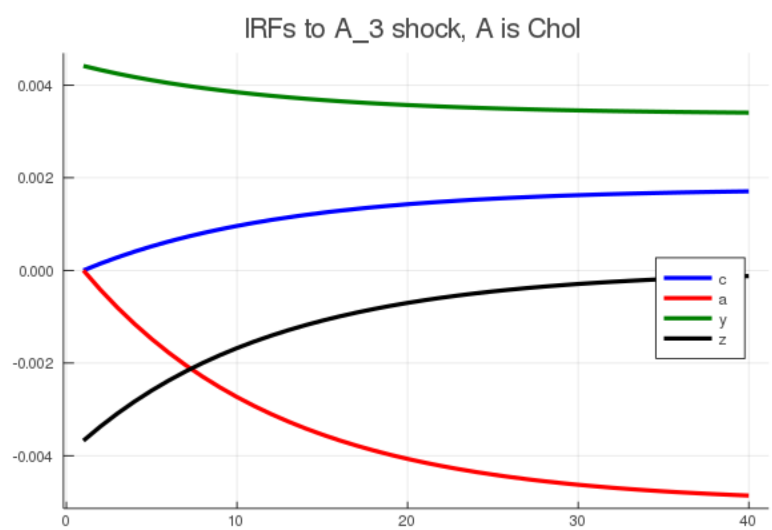
$$
\begin{bmatrix}
0.003998863840846667 & 0.0 & 0.0 \\
0.003025925151105074 & 0.009886470200427013 & 0.0 \\
0.002426272639911723 & 0.00013001968644725995 & 0.0044129950375272195
\end{bmatrix}
$$

The IRFs are below. The idea here is that we are shocking standard deviations, and using the identifying assumption of the ordering of shocks. In this case, we are specifying that when consumption is shock, everything can move contemporaneously, but when assets are shocked, only income can move at the same time, and when just income is shocked, only income moves first. Note that the ordering of the variables *really* matters for economic interpretation. I believe Chris Sims has written pretty extensively trying to figure out how we should order certain variables so that the Cholesky decomposition does what we want.

Note: For both the Cholesky and Blanchard-Quah decompositions, I was alerted to a slight scaling error in the $\hat{\Sigma}$ I used for these calculations. Upon correction, none of the plots changed beyond perhaps an multiple of 2 on the vertical axis. It seemed silly to replace six plots with no qualititave difference, but this is the reason there could be a slight discrepancy.


IRFs to A_1 shock, A is Chol
0.004
0.003
0.002
0.001
0.000
0
10
20
30
40
c
a
y
z

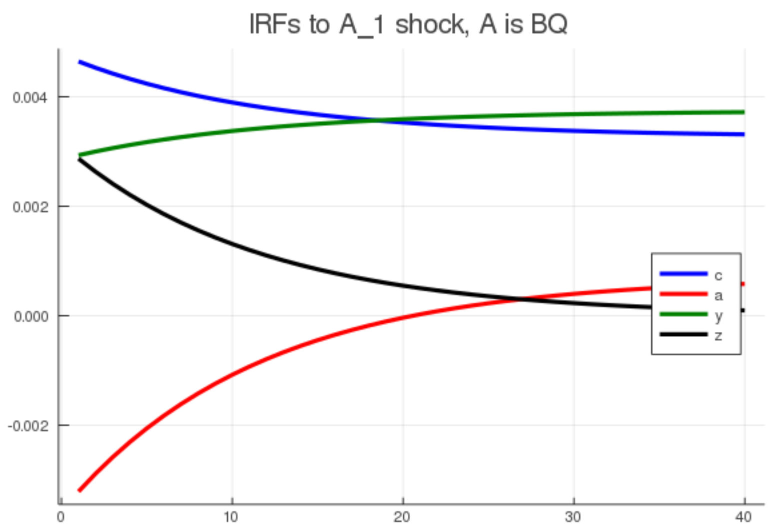

IRFs to A_2 shock, A is Chol
0.009
0.006
0.003
0.000
0
10
20
30
40
c
a
y
z

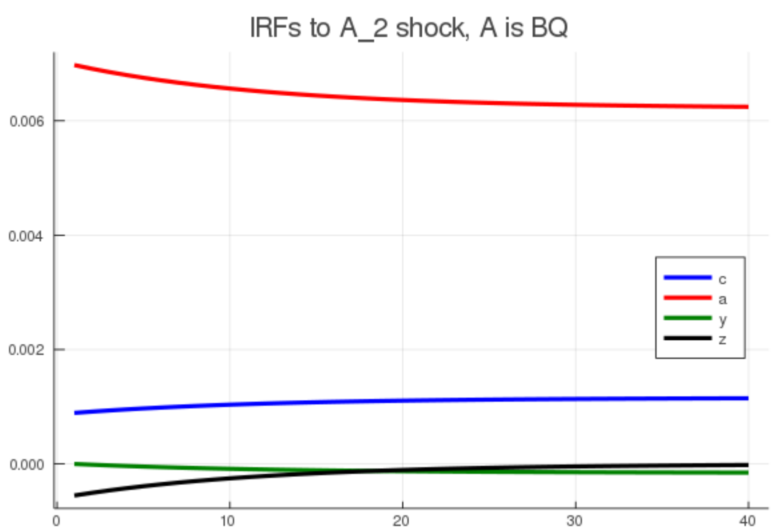

IRFs to A_3 shock, A is Chol
0.004
0.002
0.000
-0.002
-0.004
0
10
20
30
40
c
a
y
z

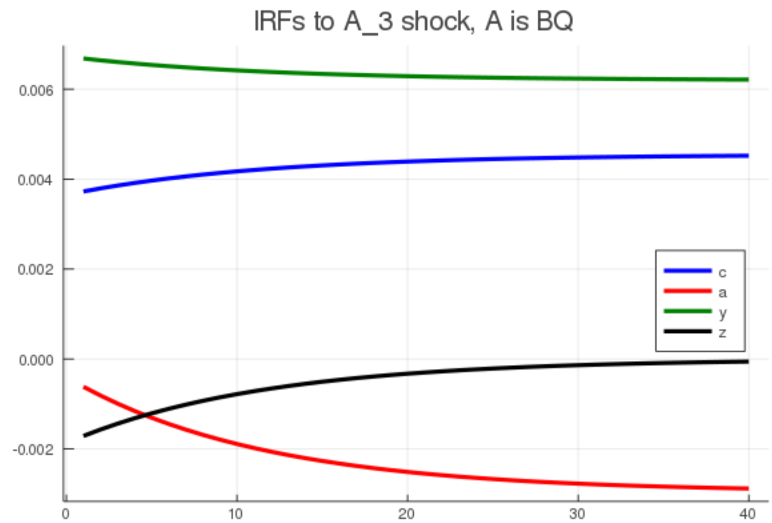
4.11

. . . .

The idea of Blanchard-Quah is that we are decomposing the shock vectors of A into transitory shocks, which die out, and permanent shocks, which do not die out. This sounds quite similar to the idea we have when decomposing B above. I have coded up what I believe to be the correct Blanchard-Quah decomposition, imposing additionally that income does not move with the second column shock, and that consumption is always positively shocked.







5 Julia Code Appendix

```
using Pkg
Pkg.add("CSV")

Pkg.add("LinearAlgebra")
using LinearAlgebra

Pkg.add("Plots")
using Plots

Pkg.add("Statistics")
using Statistics

# Plot bound
bound = pi
# Increment
dw = 0.001
omega = collect(0:dw:bound)

# Model parameters
rho = 0.9
sigma2 = 1
theta = 0.91

Sy = [sigma2/(2*pi)*(1-2*theta*cos(w)+theta^2)/(1-2*rho*cos(w)+rho^2) for w in omega]

# Plot spectrum of y
plt = plot(omega, Sy, label = string("S_y(omega), theta=",theta), legend = :right, linecolor=":

# Change theta
```

```
theta = 0.89
Sy = [sigma2/(2*pi)*(1-2*theta*cos(w)+theta^2)/(1-2*rho*cos(w)+rho^2) for w in omega]

plot!(omega, Sy, label = string("S_y(omega), theta=",theta), legend = :right, linecolor="blue"
display(plt)
# savefig("spec.png")

# Bound for plot and integration
bound = 1
# Increment
dw = 0.001
omega = collect(0:dw:bound)

# Model/filter Parameters
rho = 0.9
sigma2 = 1
omegabar = 0.2

# Indicates if x > y
function indic(x,y)
    if x > y
        return 1
    else
        return 0
    end
end

# Get spectra for x and h
Sx = [sigma2/(1-2*rho*cos(w)+rho^2)*1/(2*pi) for w in omega]
h = [indic(w,omegabar)/(2*pi) for w in omega]

# Get product spectrum
Sy = 2*pi*Sx.*h

# Plot spectrum of y
plt = plot(omega, Sy, label = "S_x(omega)", legend = :right, linecolor="red", title = "Mr. Spe
display(plt)
# savefig("mrspectrum.png")

# Find variance of y by integrating spectrum
var = 0
for w in Sy
    var += w*dw
end

# Account for symmetry
var= 2*var
```

```
# Number of leads/lags
n = 10

# Computes sum of k_j cos(w j)
function hnsum(m,w)
    sum = (pi-omegabar)/pi
    for j in 1:m
        sum += -2/(j*pi)*sin(omegabar*j)*cos(w*j)
    end
    return sum/(2*pi)
end

# Do each lag length
for i in 1:3
    hn = [abs(hnsum(i*n,w)) for w in omega]
    plt = plot(omega, hn, label = string("h^(",i*n,")(omega)"), legend = :right, linecolor="red
    plot!(omega, h, label = "h(omega)", legend = :right, linecolor="blue", title = "Ms. Spectru
    display(plt)
    savefig(string("Approxh^(",i*n,").png"))
end

using CSV
data = CSV.read("/Users/chaseabram/Downloads/cay_current_clean.csv")

# Format
data = convert(Matrix, data)
# Quarter index
quarters = data[:,1]
# c, a, y
x = data[:, 2:4]
# cay
cay = data[:,5]

# x_t and x_{t-1}
xt = x[2:size(x)[1], :]
xtm = x[1:size(x)[1]-1,:]

# Means of data
mu = [mean(xt[:,1]) mean(xt[:,2]) mean(xt[:,3])]

# constant vector
iota = ones(size(xtm)[1])

# append constant
xtm = [xtm';iota']'

# Augmented B, due to constant
Baug = inv(xtm'*xtm)*xtm'*xt
```

```
# Get errors
u = xt-xtm*Baug

# Find Sigma
Sigma = 1/(size(u)[1]-3)*u'*u
println("Sigma: ", Sigma)

# println("length(u): ", size(u)[1])

# Only consider B now
Bhat = Baug[1:3,:]

# FIX BAD TRANSPOSE FROM OLS
# We want y = Bx (Uhlig), but we ran y = xB (Shaikh)
Bhat = Matrix(Bhat')
mub = Baug[4,:]
# println("Bhat: ", Bhat)
# println("mub: ",mub)

# Jordan decomp of Bhat = V Dhat V^{-1}
Dvals = eigvals(Bhat)
V = eigvecs(Bhat)
# println("V: ",V)
Dhat = Diagonal(Dvals)
# println("Dhat: ",Dhat)

# Spot Check
# println(V*Dhat*inv(V))
# println(Bhat)

# Make unit roots true
D = Dhat
D[2,2] = 1.0
D[3,3] = 1.0

# Construct new B estimator
B = V*D*inv(V)

# Construct beta
beta = inv(V)[1,:]
# Normalize
betascale = beta[1]
beta = 1/betascale*beta
println("beta: ",beta)

# Construct betastar, nu, nustar
betastar = inv(V)[2:3,:]'
```

```
println("betastar: ",betastar)
nu = V[:,1]
println("nu: ",nu)
nustar = V[:,2:3]
println("nustar: ",nustar)

# Spot check dimensions
# println(size(beta))
# println(size(betastar))
# println(size(nu))
# println(size(nustar))

# Diagonal submatrix of stable eigenvalues
Dr = D[1,1]

# Construct alpha
alpha = nu*(I - Dr)
# Normalize with beta scale (this makes alpha*beta' = I - B)
alpha = alpha*betascale
println("alpha: ",alpha)

# Check
println("alpha*beta': ", (alpha*beta'))
println("I-B: ", I-B)


# Define z
z = xt*beta

# Plot demeaned z, so as to match cay
dxt = [z[i] - (mu*beta)[1] for i in 1:size(xt*beta)[1]]
plt = plot(quarters[2:end], dxt, label = string("z"), legend = :topright, linecolor="red", line

# Add cay
plot!(quarters[2:end], cay[2:end], label = string("cay"), linecolor="blue", linewidth = 4)
display(plt)
# savefig("cayvsz.png")

# Break up x
c = data[:,2]
iay = [ones(length(c)) data[:,3:4]]

# Run OLS
reg = inv(iay'*iay)*iay'*c

# Horizon
nq = 40
```

```
# Play with these A settings for parts 8 - 11

# Impulse matrix I
A = [1 0 0; 0 1 0; 0 0 1]

# Impulse matrix cholesky
A = cholesky(Hermitian(Sigma)).L
# println("chol(A)", A)

# Impulse matrix Blanchard-Quah
A = inv(V)*A*V
# kill the income shock from second column
A[3,2] = 0
# println(A)


# Initialize irfs
irf = zeros(3,4,nq)

# Generate responses
for i in 1:3
    # Shock
    irf[i,1:3,1] = A[:,i]
#     irf[i,4,1] = -(mu*beta)[1] + beta'*irf[i,1:3,1]
    irf[i,4,1] = beta'*irf[i,1:3,1]
    for k in 2:nq
        # With trend
#         irf[i,1:3, k] = mub + B*irf[i,1:3,k-1]
#         irf[i,4,k] = -(mu*beta)[1] + beta'*irf[i,1:3,k]

        # Without trend (looks more sensible)
        irf[i,1:3, k] = B*irf[i,1:3,k-1]
        irf[i,4,k] = beta'*irf[i,1:3,k]
    end
end

plt = []

for i in 1:3
    if i == 1
        tit = "IRFs to A_1 shock"
    elseif i == 2
        tit = "IRFs to A_2 shock"
    else
        tit = "IRFs to A_3 shock"
    end
    tit = string(tit, ", A is BQ")
    push!(plt, plot(1:nq, irf[i,1,:], label = string("c"), legend = :right, linecolor="blue", l
```

```
    plot!(1:nq, irf[i,2,:], label = string("a"), legend = :right, linecolor="red", linewidth =
    plot!(1:nq, irf[i,3,:], label = string("y"), legend = :right, linecolor="green", linewidth
    plot!(1:nq, irf[i,4,:], label = string("z"), legend = :right, linecolor="black", linewidth
#     savefig(string("BQ ", tit,".png"))
    display(plt[i])
end
```