Economics 6003 Quantitative Economics

1. Univariate ARMA Models
2. Representation Theorems
3. Data Filters

John Bluedorn

Lecturer, Economics Division

Monday/Thursday, 9/12 March 2009

Building block of ARMA models

- A(uto)R(egressive)M(oving)A(verage) = ARMA process. If there are no AR terms, it's an MA process. If there are no MA terms, it's an AR process. The mixture of the two is ARMA.
- The *white noise* stochastic process is the building block for ARMA processes. Let $\{\varepsilon_t\}$ denote a general white noise process. It always has the properties:
  1. $E(\varepsilon_t) = 0 \rightarrow$ mean-zero.
  2. $E(\varepsilon_t \varepsilon_{t-h}) = Cov(\varepsilon_t, \varepsilon_{t-h}) = 0 \forall h \neq 0 \rightarrow$ serially *un*correlated.
  3. $E(\varepsilon_t^2) = \sigma^2 \rightarrow$ finite and homoskedastic (constant variance).
- Sometimes the uncorrelated (2nd) property is replaced with independence (a much stronger property).

Building block of ARMA models

- It is also not uncommon to impose Gaussianity (normality) on the white noise process:

$$\varepsilon_t \sim NID\left(0, \sigma^2\right),$$

where *NID* denotes normally, identically distributed. Note that independence arises for free here, since zero correlation plus Gaussianity implies independence. For now, we'll not impose this.
- So far, not too interesting. We can't use the past of the white noise process to predict its future at all.
- However, if we consider *linear combinations* of the white noise process measured at different points in time, things become more interesting. We can then use past white noise elements to predict future values for the linear combination.

Moving Average (MA) models

- An MA process is a linear combination of white noise processes which advances through time (the moving part of the name).
- The shorthand for it is MA(#) where # is the *order* of the process and indicates how many lags of the past white noise affect the present value of the MA process. Thus, MA(0) is just usual white noise.
- An MA(1) process contains the current white noise process value plus the once-lagged white noise value:

$$Y_t = \mu + \varepsilon_t + \theta\varepsilon_{t-1}$$

where μ and θ are constant parameters and ε_t is the white noise process.

Moving Average (MA) models

- Notice that:
  1. $E(Y_t) = \mu$, since ε is mean-zero. If we condition on the information set at time $t-1$ however, we would include the ε_{t-1} term.
  2. $$\begin{aligned} Var(Y_t) &= E\left[(\varepsilon_t + \theta\varepsilon_{t-1})^2\right] \\ &= E\left[\varepsilon_t^2 + 2\theta\varepsilon_t\varepsilon_{t-1} + \theta^2\varepsilon_{t-1}^2\right] \\ &= \sigma^2 + \theta^2\sigma^2 \\ &= \left(1+\theta^2\right)\sigma^2 \end{aligned}$$
  3. Notice how the autocovariance for $h = 1$ is:
  $$\begin{aligned} \gamma_1 &= E\left[(\varepsilon_t + \theta\varepsilon_{t-1})(\varepsilon_{t-1} + \theta\varepsilon_{t-2})\right] \\ &= E\left[\varepsilon_t\varepsilon_{t-1} + \theta\varepsilon_{t-1}^2 + \theta\varepsilon_t\varepsilon_{t-2} + \theta^2\varepsilon_{t-1}\varepsilon_{t-2}\right] \\ &= \theta\sigma^2 \end{aligned}$$
  where we use the uncorrelatedness of the white noise to simplify.

Moving Average (MA) models

- Furthermore, all autocovariances for $h > 1$ are zero, by the same uncorrelatedness.
- This is extremely nice. We get that an MA(1) process is weakly stationary just from its structure. Moreover, if we make the additional assumption of Gaussianity of the white noise process, we get ergodicity for all moments.
- In general, an MA(q) process, where q is finite and positive, is always weakly stationary, *regardless* of the magnitude of the coefficients on the MA terms. Why? All first and second moments are finite and well-defined. Note that the autocovariances for $h > q$ are all zero by uncorrelatedness of the white noise.
- What about the case where q is infinite? As you would expect, things get trickier and we have to impose more assumptions in order for things to converge.

Moving Average (MA) models

- Consider the MA(∞) process:

$$Y_t = \mu + \sum_{j=0}^{\infty} \psi_j \varepsilon_{t-j}$$

- In order for the MA(∞) to be weakly stationary, we need for the MA coefficients to be *square summable*:

$$\sum_{j=0}^{\infty} \psi_j^2 < \infty$$

which ensures that the higher order covariances shrink quickly enough. By convention, $\psi_0 = 1$.

- If we go for the somewhat stronger *absolute summability*, where $\sum_{j=0}^{\infty} |\psi_j| < \infty$, then we get not only weak stationarity but also ergodicity for the mean.

Moving Average (MA) models

- Consider the derivative of the MA process with respect to a past white noise innovation:

$$\frac{\partial Y_t}{\partial \varepsilon_{t-j}} = \psi_j \forall j \geq 0.$$

- How can we interpret this? The uncorrelatedness of the εs allows us to extract the marginal effect of the innovation at time $(t - j)$ upon the MA process at time t. The MA coefficient is the *impulse response* of Y_t to ε_{t-j}.
- A ready causal interpretation presents itself if we consider the relevant counterfactual to be the average worldline where $\varepsilon_{t-j} = 0$. Then, the impulse response of an MA process to its underlying white noise process represents the causal effect of an innovation over time.
- Of course, if the model is misspecified, then this interpretation is not accurate (viz., there are omitted variables).

Moving Average (MA) models

- Consider the derivative of the MA process with respect to a past white noise innovation:

$$\frac{\partial Y_t}{\partial \varepsilon_{t-j}} = \psi_j \forall j \geq 0.$$

- How can we interpret this? The uncorrelatedness of the εs allows us to extract the marginal effect of the innovation at time $(t-j)$ upon the MA process at time t. The MA coefficient is the *impulse response* of Y_t to ε_{t-j}.
- A ready causal interpretation presents itself if we consider the relevant counterfactual to be the average worldline where $\varepsilon_{t-j} = 0$. Then, the impulse response of an MA process to its underlying white noise process represents the causal effect of an innovation over time.
- Of course, if the model is misspecified, then this interpretation is not accurate (viz., there are omitted variables).

Moving Average (MA) models

- Since the explanatory variables are unobserved (the εs), how can you estimate an MA model?
  - If you are willing to make a specific distributional assumption for the εs (e.g., Gaussianity), then you can use maximum likelihood, combined with a set of initial conditions for the εs.
    - Conditional maximum likelihood starts from the initial condition and iterates the conditional density forward in time. With the initial conditions, this is just a factorization of the usual joint density.
    - If exact likelihood method is used, then you can apply the Kalman filter or undertake a triangular factorization of the covariance matrix to get the appropriate likelihood.
  - Otherwise, you can approach the problem as quasi-maximum likelihood, where you use a misspecified likelihood as an approximation.

Autoregressive (AR) models

- An AR process is a linear combination of its own past values (the autoregressive part of the name) plus a contemporaneous innovation.
- The shorthand for it is AR(#) where # is the *order* of the process and indicates how many lags of the process affect the present value of the AR process. Thus, AR(0) is just usual white noise (only the contemporaneous innovation piece).
- An AR(1) process contains the current white noise process value plus the once-lagged value of the process:

$$Y_t = c + \phi Y_{t-1} + \varepsilon_t$$

where c and ϕ are constant parameters and ε_t is a white noise process (the innovation).

Autoregressive (AR) models

- Notice how this is essentially the first order linear difference equation, with a forcing variable added (the innovation).
- We know that for the solution to such a difference equation to depend only upon the past, we need for $|\phi| < 1$. In this case, we can "divide-out" the past Y and get an expression for the Y_t solely as a function of past values of ε and the constant. In fact, the AR(1) model is equivalent to an MA(∞) model!:

$$
\begin{aligned}
(1 - \phi L) Y_t &= c + \varepsilon_t \\
Y_t &= \frac{c}{(1-\phi)} + (1 - \phi L)^{-1} \varepsilon_t \\
&= \frac{c}{(1-\phi)} + \sum_{j=0}^{\infty} \phi^j \varepsilon_{t-j}
\end{aligned}
$$

Autoregressive (AR) models

- A couple of things to notice here:
  - The AR(1) impulse response function is just the AR coefficient exponentiated to the lag (or horizon) length:

$$\frac{\partial Y_t}{\partial \varepsilon_{t-j}} = \phi^j \forall j \geq 0.$$

    It's a smooth, geometrically declining function. This is in contrast to the general MA(∞) case, where the MA coefficients may or may not be smoothly related across time horizons.
  - $|\phi| < 1$ is also a sufficient condition for weak stationarity and ergodicity for the mean since:

$$\sum_{j=0}^{\infty} |\phi^j| < \infty$$

- What about an AR(p) model, where p is finite and positive?

Autoregressive (AR) models

- As usual, we'll look for a solution to the implied difference equation by factoring out a lag polynomial, and then looking to "divide-out". So, for the AR(p) model, we get something like:

$$\begin{aligned} Y_t &= c + \phi_1 Y_{t-1} + \phi_2 Y_{t-2} + \cdots + \phi_p Y_{t-p} + \varepsilon_t \\ &= c + \left(\phi_1 L + \phi_2 L^2 + \cdots + \phi_p L^p\right) Y_t + \varepsilon_t \Rightarrow \\ & \left(1 - \phi_1 L - \phi_2 L^2 - \cdots - \phi_p L^p\right) Y_t = c + \varepsilon_t \end{aligned}$$

- In order to be able to invert it, we need for all of the roots of the associated polynomial:

$$\left(1 - \phi_1 z - \phi_2 z^2 - \cdots - \phi_p z^p\right) = 0.$$

to be *outside the unit circle* in modulus (they might be complex numbers).

Autoregressive (AR) models

- This is equivalent to the requirement that the eigenvalues (λ here) of the characteristic polynomial defined by:

$$\left(\lambda^p - \phi_1 \lambda^{p-1} - \phi_2 \lambda^{p-2} - \cdots - \phi_p\right) = 0.$$

  lie *inside the unit circle* in modulus. The characteristic polynomial formulation arises naturally if we express the difference equation in matrix notation (we'll consider this in a moment).
- Unfortunately, both expressions are used in the literature. You have to use the context to figure out to which polynomial an author is referring.
- If these hold, then you can show that the AR(p) maps to an MA(∞) process, where the MA coefficients satisfy $\sum_{j=0}^{\infty} |\psi_j| < \infty$, meaning that the process is weakly stationary and ergodic for the mean.

Autoregressive (AR) models

- Notice that any AR(p) model can be rewritten as an AR(1) model in matrix notation:

$$\xi_t = \underset{(p\times 1)}{\begin{bmatrix} Y_t \\ Y_{t-1} \\ \vdots \\ Y_{t-p+1} \end{bmatrix}}, \quad F = \underset{(p\times p)}{\begin{bmatrix} \phi_1 & \phi_2 & \cdots & \phi_p \\ 1 & 0 & 0 & 0 \\ 0\vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 1 & 0 \end{bmatrix}},$$

$$\text{and } v_t = \underset{(p\times 1)}{\begin{bmatrix} c + \varepsilon_t \\ 0 \\ \vdots \\ 0 \end{bmatrix}} \Rightarrow \xi_t = F\xi_{t-1} + v_t, \text{ giving:}$$

$$Y_t = c + \phi_1 Y_{t-1} + \phi_2 Y_{t-2} + \cdots + \phi_p Y_{t-p} + \varepsilon_t$$

and a set of identities.

Autoregressive (AR) models

- The characteristic roots which determine whether or not the model is weakly stationary can be found from the matrix F. They solve:

$$|F - \lambda I_p| = 0$$

- We won't deal explicitly with solving characteristic roots from the matrix equation, but it is important to recognize that that is what's going on in the background.
- Another thing to remark upon is that we can move back and forth between viewing things as AR or MA. Under a weak stationarity condition on the MA(1) process, we can invert it and get an AR(∞) process (a similar argument holds for any MA(q) for q finite and positive).

ARMA models

- An ARMA model is a linear combination of an AR component with an MA component, generally denoted by ARMA(p, q). Here p is the order of the AR component and q is the order of the MA component.
- Since we know that any finite order MA is weakly stationary, the critical determinant of stationarity of the ARMA model is the stationarity of the AR component, and thus the AR lag polynomial. Why? Again, to solve and eliminate the past Ys, we have to "divide-out" the AR lag polynomial.
- To save space, an ARMA model is often written as:

$$\Phi(L) Y_t = \Theta(L) \varepsilon_t,$$

where $\Phi(L)$ is the AR lag polynomial and $\Theta(L)$ is the MA lag polynomial.

ARMA models

Some caveats when dealing with ARMA models:

- It is possible to have multiple representations of an ARMA process, if we multiply both sides by the same lag polynomial. As Hamilton notes, this can make things confusing (and screw-up any numerical optimization), so we always go with the simplest parametrization of the model.
- Related to this, if the AR and MA lag polynomials of an ARMA(p, q) process share a common root, then the process can be represented as an ARMA($p-1, q-1$), where the common root term of the polynomials is canceled out.

ARMA models

- As with usual linear difference equations, we can solve an ARMA model *even if* the AR lag polynomial in L is not well-defined. How? We rewrite it as a polynomial in L^{-1}, which will converge when inverted. Of course, this means that the present Y depends upon the future innovations!
  - If the AR lag polynomial in L is well-defined, then we say that we have the *invertible* representation of the ARMA process. It is also called the *fundamental* representation. You will also sometimes hear it called (improperly IMHO) the causal representation, since it expresses the present solely as a function of the past.
  - If the AR lag polynomial in L is not well-defined, then we say that we have the *non-invertible* representation.

We'll talk about these issues more when we discuss identification and invertibility.

ARMA in Stata

- Stata allows for direct use of lag operator notation if we declare a loaded dataset as time-series. The time-series declaration command will be something like:

```
tsset year, yearly;
```

  where year is a variable in the dataset containing the time. Note how I am using the semi-colon to denote a carriage-return. When scripting in Stata, you can set it as a delimiter if you wish.

- For an AR model, we can estimate a simple linear regression using the `tsseted` data by something like:

```
regress yvar L(1/4).yvar, vce(bootstrap,
        reps(1000) seed(123456);
```

  where yvar is the dependent variable and its own lags (4 of them here) are the explanatory variables. The `vce` term tells Stata how to calculate the variance/covariance matrix. I've opted for bootstrapping.

ARMA in Stata

- As a general rule for AR models, we want to make sure there are enough lags of the dependent variable in the model to soak up any serial correlation in the errors. Why? Serial correlation implies a failure of the basic OLS orthogonality condition for the errors and the explanatory variables.
- So, a usual practice is to pick a lag order for an AR model based upon a failure to reject the null of no serial correlation.
- The bootstrapping is undertaken here because the resulting standard errors will be robust to heteroskedasticity (Stata does a paired bootstrap).
  - It involves drawing random samples with replacement from the dataset, estimating the model from each random sample, and collecting that coefficients. This builds an empirical distribution for the estimated coefficients, which we can use to calculate a standard error.

ARMA in Stata

- We could also use the usual Huber-Eicker-White heteroskedasticity-robust standard errors via the `robust` option.
- A constant term is automatically included by Stata with `regress` unless you tell it otherwise (`noconstant`).
- Instead of adding additional lags of the dependent variable to eliminate the serial correlation, we might instead opt to model the serial correlation directly via an MA process. In this case, we need to use Stata's `arima` command to estimate. It does maximum (or quasi-maximum) likelihood estimation.
- Suppose that we want to estimate an ARMA(4,2) model. We specify it as:

```
arima yvar , ar(1/4) ma(1/2);
```

  Note that we can specify any subset of lags to include (need not be contiguous block).

ARMA in Stata

- Use `help arima` to get the full set of options which you can specify.
- We aren't considering the `i` bit here, but that stands for integrated and refers to the amount of differencing a series requires before it is stationary. We'll talk about this when we get to nonstationary time series.
- You can also have exogenous explanatory variables with `arima` and thus estimate an ARMAX model (the X are the additional exogenous explanatory variables).
- There are a couple of cool prediction options that you can do with `arima` in Stata:
  1. One-step ahead predictions post-estimation via: `predict yvarhat1, xb;.`
  2. Recursive or dynamic predictions post-estimation via: `predict yvarhat2, dynamic(.);.`

Romer and Romer (2004) example

- Romer and Romer (2004) undertake an interesting identification approach to estimate the effect of U.S. monetary policy upon U.S. economic performance (as measured by the growth of real output and inflation).
  - They use narrative evidence (minutes and transcripts) of the Federal Reserve's Federal Open Market Committee (FOMC) to construct an intended federal funds (FF) rate series, capturing the FOMC's target interest rate.
  - They then employ the in-house economic forecasts of the Federal Reserve as explanatory variables for changes in the intended FF rate.
  - They argue that the residuals from such a regression represent the change in monetary policy intentions which is *unrelated* to expected future economic conditions, and is therefore *exogenous*.

Romer and Romer (2004) example

- Specifically, they estimate:

$$\begin{aligned}\Delta ff_m &= \alpha + \beta ff_{m-1} \\ &+ \sum_{j=-1}^{2} \gamma_j \widehat{\Delta y}_{m,j} + \sum_{j=-1}^{2} \eta_j \left(\widehat{\Delta y}_{m,j} - \widehat{\Delta y}_{m-1,j} \right) \\ &+ \sum_{j=-1}^{2} \theta_j \hat{\pi}_{m,j} + \sum_{j=-1}^{2} \lambda_j \left(\hat{\pi}_{m,j} - \hat{\pi}_{m-1,j} \right) \\ &+ \mu \hat{n}_{m,0} + \eta_m,\end{aligned}$$

where m indexes FOMC meetings, j indexes the forecast quarter relative to the current meeting's quarter, ff is the target federal funds rate level, Δy is real output growth, π is inflation, n is the unemployment rate, η is a mean-zero error term, and a hat denotes the real-time forecast for a variable.

Romer and Romer (2004) example

- The values of $\hat{\eta}$ are then cumulated within months to generate a monthly data series stretching from 1969-1996. This is then used to estimate the response of real output growth (and other variables) at the monthly frequency.
- They are essentially arguing that their procedure identifies a set of exogenous monetary policy shocks that are orthogonal to the relevant counterfactual world, where monetary policy is purely endogenous. Notice how this implies that an impulse response, where the $\hat{\eta}$ are the innovations, is then the appropriate way to evaluate the causal effect of monetary policy.
- They estimate the following ARX(24,36) model:

$$y_t = \sum_{j=1}^{12} \alpha_j D_j + \sum_{j=1}^{24} \beta_j y_{t-j} + \sum_{j=1}^{36} \gamma_j \widehat{\eta_{t-j}} + u_t$$

D are monthly dummies.

Romer and Romer (2004) example

TABLE 3—THE IMPACT OF MONETARY POLICY SHOCKS ON INDUSTRIAL PRODUCTION

Monetary policy shock			Change in industrial production		
Lag	Coefficient	Standard error	Lag	Coefficient	Standard error
1	0.0038	0.0018	1	0.063	0.064
2	0.0026	0.0018	2	−0.013	0.063
3	−0.0038	0.0018	3	0.107	0.063
4	−0.0012	0.0018	4	0.048	0.063
5	−0.0039	0.0018	5	0.028	0.063
6	−0.0001	0.0018	6	−0.005	0.063
7	−0.0008	0.0019	7	0.018	0.063
8	−0.0029	0.0019	8	0.008	0.063
9	−0.0021	0.0019	9	0.040	0.062
10	−0.0047	0.0018	10	−0.043	0.061
11	−0.0025	0.0019	11	0.071	0.059
12	−0.0035	0.0019	12	0.287	0.060
13	−0.0021	0.0019	13	0.023	0.061
14	−0.0007	0.0018	14	−0.196	0.060
15	−0.0003	0.0019	15	−0.151	0.061
16	0.0019	0.0018	16	−0.128	0.062
17	−0.0009	0.0018	17	0.078	0.063
18	−0.0024	0.0018	18	0.085	0.063
19	−0.0023	0.0019	19	0.056	0.063
20	−0.0007	0.0019	20	0.081	0.063
21	−0.0011	0.0019	21	−0.060	0.063
22	−0.0032	0.0018	22	−0.017	0.063
23	0.0015	0.0019	23	−0.068	0.063
24	−0.0000	0.0019	24	0.086	0.063
25	−0.0001	0.0019			
26	−0.0000	0.0019			
27	−0.0007	0.0019			
28	0.0038	0.0019			
29	0.0013	0.0019			
30	0.0035	0.0019			
31	0.0018	0.0019			
32	0.0009	0.0018			
33	0.0014	0.0018			
34	0.0047	0.0018			
35	0.0011	0.0018			
36	0.0024	0.0018			

Notes: $R^2 = 0.86$; D.W. = 2.01; s.e.e. = 0.009; N = 324. The sample period is 1970:1–1996:12. Coefficients and standard errors for the constant term and monthly dummies are not reported.

Romer and Romer (2004) example

- The raw coefficients from the regression are difficult to interpret, since effects are also affecting the lagged dependent variables. There is also a surfeit of them – too many numbers! The impulse response to an innovation in the monetary policy shock is more useful.
- Why do they chose this lag specification? They argue that there are important direct effects which are not captured in the lagged dependent variable terms. The exact choice of 24 and 36 are otherwise arbitrary, decided upon using a heuristic of 2 years indirect with 3 years direct effect of monetary policy.
- Notice how the Durbin-Watson statistic is essentially 2, indicating no AR(1) serial correlation. This is not surprising, given how many lags there are. However, there could be higher order serial correlation; DW doesn't say anything about that.

Romer and Romer (2004) example - IRFs

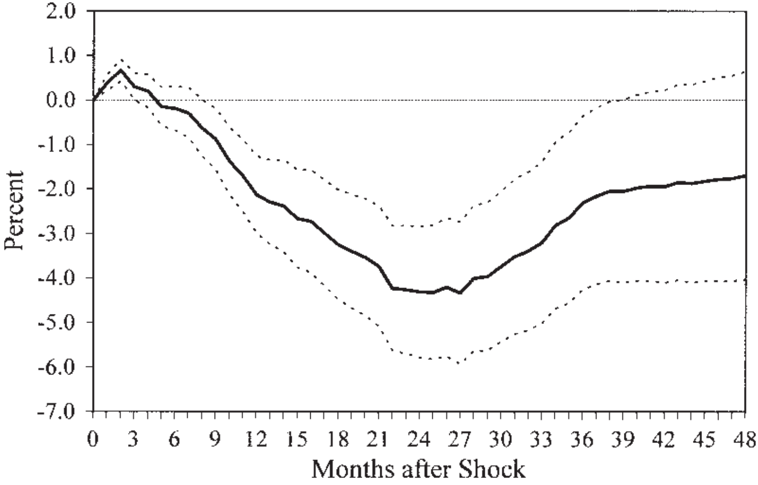
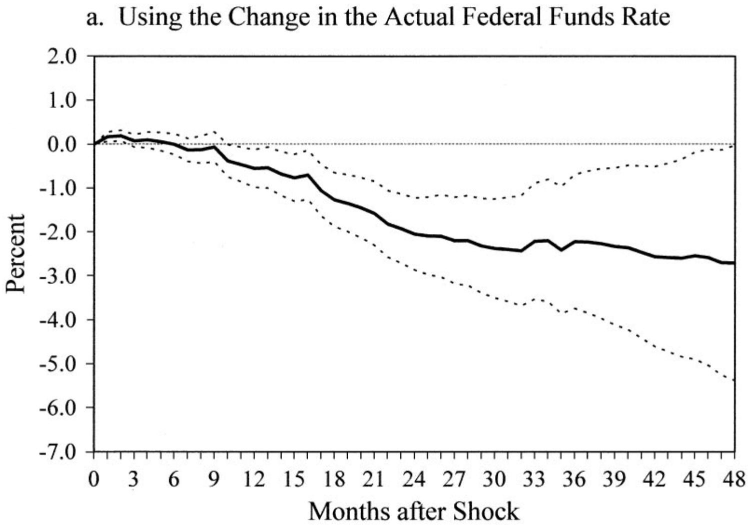


FIGURE 2. THE EFFECT OF MONETARY POLICY ON OUTPUT

Romer and Romer (2004) example - IRFs



Romer and Romer (2004) example

- Other than VAR estimation, there is no canned routine in Stata which will generate impulse response functions.
- Consequently, you have to program something yourself:
    - Use `nlcom` for non-linear combinations to directly calculate the impulse responses and their delta-method (asymptotic, analytic) standard errors.
    - Place the estimated coefficients in a matrix and use the MATA language within Stata to "power-up" the matrix representation of the system to get the impulse responses. For standard errors, you would: (1) generate lots of bootstrap samples, get the coefficient estimates, and export them; (2) calculate the delta-method, analytical standard errors.
    - Export the estimated coefficients for use in another program (e.g., Matlab). For standard errors, the same options as above are available.

Representation theorems

- There are two representation theorems for weakly stationary time series that are useful to know. We won't derive them, but will merely state them and discuss their interpretation. The first is due to Wold (1938) and the second due to Cramér (1942).
- **Wold Decomposition Theorem** – *Any* zero-mean, weakly stationary process (denote it Y_t) can be expressed as the linear combination of a stochastic component (a linear combination of a white noise process and its lags) and a deterministic component that is uncorrelated with the stochastic component:

$$Y_t = \sum_{j=0}^{\infty} \psi_j \varepsilon_{t-j} + \kappa_t, \text{ where } \psi_0 = 1, \sum_{j=0}^{\infty} \psi^2 < \infty.$$

Wold Decomposition Theorem

- ε_t is a white noise process. Moreover, it is the prediction error for a linear projection of Y_t upon its own past.
- $corr(\varepsilon_s, \kappa_t) = 0 \forall s, t$.
- κ_t is linearly deterministic. It can be perfectly linearly predicted by lags of Y_t.
- Things to note:
  - The mean-zero is not a real restriction, since we can always redefine the dependent variable as Y minus its mean.
  - εs do *not* have to be Gaussian. Moreover, they do *not* have to be independent or identically distributed.
  - Although the εs are linearly orthogonal to the past Y, it does *not* have to be the case that their conditional expectation on the past Y is zero (could be *non*-linearly related).
  - εs do *not* necessarily have *any* structural interpretation.
  - The Wold representation is the *unique linear* representation with *linear forecast* errors. There may be other *non*-linear representations or *non*-linear forecast error representations.

Cramér Representation Theorem

- **Cramér Representation Theorem** (also known as the *spectral representation theorem* – Any weakly stationary stochastic process with absolutely summable autocovariances (related to the MA coefficients) can be expressed in the form:

$$Y_t = \mu + \int_{-\pi}^{\pi} e^{i\omega t} dZ_Y(\omega)$$

where $i = \sqrt{-1}$, ω is frequency in radians, and $dZ_Y(\omega)$ is a mean-zero, complex-valued random variable that is continuous in ω.

  - dZ is *uncorrelated* across frequencies. Thus, you can decompose the variation in Y_t into the variation due to cycles of different frequencies.
  - This is closely related to the Fourier transform of the autocovariances (known as the *spectral density* of the process).

Data Filters

- A lot of empirical work relies upon macroeconomic data that are filtered. This means that some component of the variability of the data is removed. Why would you want to do this?
  - We want to focus on some aspect of the data without getting distracted by other irrelevant aspects of the data.
  - The classic example is filtering to abstract away from business cycle fluctuations.
- The Cramér representation theorem gives us a ready-way to filter out the business cycle component, by removing those frequencies of the data associated with business cycles.
  - Determine the data frequency (e.g., monthly, quarterly, yearly, etc.).
  - Determine what are the relevant business cycles. Baxter and King (1999) consider them to be recurrent fluctuations between 1.5-8 years.
  - Determine the corresponding frequency ω in radians $\rightarrow \omega = \frac{2\pi}{\tau}$, where τ is the time cycle. Notice how long

Data Filters

- A lot of empirical work relies upon macroeconomic data that are filtered. This means that some component of the variability of the data is removed. Why would you want to do this?
  - We want to focus on some aspect of the data without getting distracted by other irrelevant aspects of the data.
  - The classic example is filtering to either abstract away from business cycle fluctuations, or to abstract away from *non*-business cycle fluctuations (e.g., the very short and long runs).
- The Cramér representation theorem gives us a ready-way to filter out the cyclical component we don't want, by removing those frequencies of the data associated with those cycles.

Data Filters

- How can we figure out what frequencies correspond to the cycle of interest? The procedure goes as follows:
  - Determine the data frequency (e.g., monthly, quarterly, yearly, etc.).
  - Determine what are the relevant business cycles. Baxter and King (1999) consider them to be recurrent fluctuations between 1.5-8 years. Hodrick and Prescott (1997) consider them to be all recurrent fluctuations below about 8 years.
  - Determine the corresponding frequency ω in radians $\rightarrow \omega = \frac{2\pi}{\tau}$, where τ is the cycle in the data's time units (e.g., months, quarters, etc.). Notice how long cycles correspond to low frequencies and short cycles correspond to high frequencies.
  - Set the filter to eliminate those frequencies that are not of interest.

Data Filters

- For most economic work, this all boils down to choosing one of the two most popular filters:
  - Hodrick and Prescott (1997) – filters out all frequencies either above or below a given threshold.
  - Baxter and King (1999) – filters out or preserves those frequencies between some upper and lower bound.
- We won't go into their derivation, but you can execute these filters on `tsset` time series data in Stata by installing the `hprescott` and `bking` user-defined packages $\rightarrow$ on the command line in Stata, type `net search hprescott` and then click on the appropriate link (for example).
- You will sometimes hear them referred to as *high-pass* filters, since the higher, business cycle frequencies are passed or kept.

Data Filters

- There are several caveats to using filters:
  - If you are doing work that is implicitly *real-time* (e.g., modeling information sets at a point in time or the like), then you need to use the *one-sided* version of the filters. By default, they use information from both the future and the past to locate the cycles and then pass or eliminate them; they are *two-sided* filters.
  - As Cogley (2008) describes, the Baxter-King approach defines business cycles as linearly deterministic (recall the Wold decomposition). This does not accord with the view from DSGE models which sees the business cycle as a result of stochastic processes which propagate.
  - Cogley and Nason (1995) showed that the application of the Hodrick-Prescott filter to a random walk creates a spurious business cycle! Thus, the filter can actually generate cycles when they are not there in an economically meaningful way.

Data Filters

- Cogley (2008) notes that interest in business cycle research is now less on fitting moments to filtered business cycle data than on matching impulse response functions.
- We will discuss what this actually entails and how it relates to identification issues when we get to V(ector) A(uto)R(egressions) – VARs.

Summary

- An ARMA process may be inverted to get an MA process, where the MA coefficients represent the impulse response to the corresponding innovation.
- MA processes always have 2 representations, only one of which is considered *fundamental*. It is where the present is solely a function of the past (generally preferred).
- For highly parameterized ARMA models, the impulse response functions are oftentimes more revealing than the actual estimated coefficients. Of course, their interpretation depends upon what the relevant counterfactual is.
- The Wold decomposition theorem guarantees that we can always find an MA representation. However, it says nothing about whether or not the resulting representation is structural or economically meaningful.

Summary

- The Cramér representation theorem gives us a nice link between the time domain (where we usually work) and the frequency domain, which is useful for business cycle research and filters.
- A naive application of a filter can actually generate a spurious business cycle (a cycle which has no economic meaning). This is related to the Wold decomposition not necessarily having any structural interpretation.