Modeling Stationary Time Series

Christopher Adolph

Department of Political Science

and

Center for Statistics and the Social Sciences

University of Washington, Seattle
The story so far

We’ve learned:

• how to decide whether one estimator is “better” than another under a given DGP

• why our LS models don’t work well with time series

• how to obtain quantities of interest, such as $\mathbb{E}(y|x_c)$ from an estimated model

• the basics of time series dynamics, including:
  trends, autoregression, moving averages, seasonality, stationarity
What we’re doing today

Next steps:

• Use ML to estimate AR($p$), MA($q$), and ARMA($p,q$) models for stationary series

• Use our time series knowledge & MLE fitting tools to select $p$ and $q$

• Use simulations to understand how $\mathbb{E}(y_t|x_t)$ changes as we vary $x_t$ over time
An AR(1) Regression Model

To create a regression model for an AR(1) process, we allow the mean of the process to shift by adding $c_t$ to the equation:

$$y_t = y_{t-1} \phi_1 + c_t + \varepsilon_t$$

We then parameterize $c_t$ as the sum of a set of time varying covariates, $x_{1t}, x_{2t}, x_{3t}, \ldots$

and their associated parameters, $\beta_1, \beta_2, \beta_3, \ldots$

which we compactly write in matrix notation as $c_t = x_t \beta$
An AR(1) Regression Model

Substituting for $c_t$, we obtain the AR(1) regression model:

$$y_t = y_{t-1} \phi_1 + x_t \beta + \varepsilon_t$$

Estimation is by maximum likelihood, *not* LS

(We will discuss the LS version later)

MLE accounts for dependence of $y_t$ on past values; complex derivation (see James Hamilton, *Time Series Analysis* for a review)

We’ll focus on interpreting this model in practice
Aside: the AR(1) likelihood function

Why is the MLE for AR(1) more complex than the MLE for linear regression?

Suppose our time series “starts” at \( t = 1 \):
there is no lag before \( t = 1 \), so period 1 has no AR(1) term

Then the distribution of the first observation is

\[
y_1 \sim \mathcal{N}(x_1 \beta, \sigma^2)
\]

But after \( t = 1 \), \( y_t \) is AR(1), so \( y_{t+1} \) depends on \( y_t \)

\[
y_2 | y_1 \sim \mathcal{N}(x_2 \beta + \phi y_1, \sigma^2)
\]

\[
y_3 | y_2 \sim \mathcal{N}(x_3 \beta + \phi y_2, \sigma^2)
\]

and so on up to the distribution of \( y_t \)

This means the \( y_t \)'s are not iid: the usual Normal MLE is inadequate,
and we must create a new likelihood based on the distributions above
Aside: the AR(1) likelihood function

Multiplying together the pdfs of the distributions of \( y_1, \ldots y_t \) and reducing to sufficient statistics yields the following log-likelihood for AR(1):

\[
\mathcal{L}(\beta, \phi_1 | y, X) = -\frac{1}{2} \log \left( \frac{\sigma^2}{1 - \phi_1^2} \right) - \frac{\left( y_1 - \frac{x_1 \beta}{1 - \phi_1} \right)^2}{2\sigma^2} - \frac{T - 1}{2} \log \sigma^2 - \sum_{t=2}^{T} \frac{(y_t - x_t \beta - \phi_1 y_{t-1})^2}{2\sigma^2}
\]

Only differs from least squares in the treatment of \( y_1 \), so very similar to OLS with a lagged DV if \( T \) is large

But LS standard errors can be substantially biased if \( T \) is small

The definition of “small” depends on \( \phi, \sigma \), and covariates, so you try both the AR(1) MLE and OLS if you are worried!
Aside: the AR(1) likelihood function

Multiplying together the pdfs of the distributions of \( y_1, \ldots, y_t \) and reducing to sufficient statistics yields the following log-likelihood for AR(1):

\[
\mathcal{L}(\beta, \phi_1 | y, X) = -\frac{1}{2} \log \left( \frac{\sigma^2}{1 - \phi_1^2} \right) - \frac{\left( y_1 - \frac{x_1 \beta}{1 - \phi_1} \right)^2}{2\sigma^2} - \frac{T - 1}{2} \log \sigma^2 - \sum_{t=2}^T \frac{(y_t - x_t \beta - \phi_1 y_{t-1})^2}{2\sigma^2}
\]

MLEs only get more complex as we move towards ARMA(p,q)

Generally, we can treat ARMA estimation as a black box

Our main concern will be how to select the right model and interpret what it means substantively
Interpreting AR(1) parameters

Suppose that a country’s GDP follows this simple model

\[ \text{GDP}_t = \phi_1 \text{GDP}_{t-1} + \beta_0 + \beta_1 \text{Democracy}_t + \varepsilon_t \]
\[ \text{GDP}_t = 0.9 \times \text{GDP}_{t-1} + 10 + 2 \times \text{Democracy}_t + \varepsilon_t \]

Suppose that at year \( t \), \( \text{GDP}_t = 100 \),
and the country is a non-democracy, \( \text{Democracy}_t = 0 \).

What would happen if we “made” this country a democracy in period \( t + 1 \)?
Interpreting AR(1) parameters

\[ y_t = y_{t-1} \phi_1 + x_t \beta + \varepsilon_t \]

Recall:
an AR(1) process can be viewed as the geometrically declining sum of all its past errors.

When we add the time-varying mean \( x_t \beta \) to the equation, the following now holds:

\[ y_t = (x_t \beta + \varepsilon_t) + \phi_1 (x_{t-1} \beta + \varepsilon_{t-1}) + \phi_1^2 (x_{t-2} \beta + \varepsilon_{t-2}) + \phi_1^3 (x_{t-3} \beta + \varepsilon_{t-3}) + \ldots \]

That is, \( y_t \) represents the sum of all past \( x_t \)'s as filtered through \( \beta \) and \( \phi_1 \)
Interpreting AR(1) parameters

Take a step back:
suppose \( c_t \) is actually fixed for all time at \( c \),
so that \( c = c_t \)

Now, we have

\[
y_t = (c + \varepsilon_t) + \phi_1(c + \varepsilon_{t-1}) + \phi_1^2(c + \varepsilon_{t-2}) + \phi_1^3(c + \varepsilon_{t-3}) + \ldots
\]

\[
= \frac{c}{1 - \phi_1} + \varepsilon_t + \phi_1 \varepsilon_{t-1} + \phi_1^2 \varepsilon_{t-2} + \phi_1^3 \varepsilon_{t-3} \ldots
\]

which follows from the limits for infinite series

Taking expectations removes everything but the first term:

\[
\mathbb{E}(y_t) = \frac{c}{1 - \phi_1}
\]

Implication:
if, starting at time \( t \) and going forward to \( \infty \),
we fix \( x_t\beta \),
then \( y_t \) will converge to \( x_t\beta/(1 - \phi_1) \)
Interpreting AR(1) parameters

\[
\begin{align*}
\text{GDP}_t &= \phi_1 \text{GDP}_{t-1} + \beta_0 + \beta_1 \text{Democracy}_t + \varepsilon_t \\
\text{GDP}_t &= 0.9 \times \text{GDP}_{t-1} + 10 + 2 \times \text{Democracy}_t + \varepsilon_t
\end{align*}
\]

If at year \( t \), \( \text{GDP}_t = 100 \) and the country is a non-democracy (\( \text{Democracy}_t = 0 \)) then:

This country is in a steady state – it will tend to have GDP of 100 every period, with small errors from \( \varepsilon_t \) (verify this)
Interpreting AR(1) parameters

\[
\begin{align*}
\text{GDP}_t &= \phi_1 \text{GDP}_{t-1} + \beta_0 + \beta_1 \text{Democracy}_t + \varepsilon_t \\
\text{GDP}_t &= 0.9 \times \text{GDP}_{t-1} + 10 + 2 \times \text{Democracy}_t + \varepsilon_t
\end{align*}
\]

Now suppose we make the country a democracy in period \( t + 1 \), so that \( \text{Democracy}_{t+1} = 1 \).

The model predicts that in period \( t + 1 \), the level of GDP will rise by \( \beta = 2 \), to 102.

This \textit{appears} to be a small effect, but...
Interpreting AR(1) parameters

\[ \text{GDP}_t = \phi_1 \text{GDP}_{t-1} + \beta_0 + \beta_1 \text{Democracy}_t + \varepsilon_t \]
\[ \text{GDP}_t = 0.9 \times \text{GDP}_{t-1} + 10 + 2 \times \text{Democracy}_t + \varepsilon_t \]

... the effect accumulates, so long as \text{Democracy} = 1

\[ \mathbb{E}(\hat{y}_{t+2}|x_{t+2}) = 0.9 \times 102 + 10 + 2 = 103.8 \]
\[ \mathbb{E}(\hat{y}_{t+3}|x_{t+3}) = 0.9 \times 103.8 + 10 + 2 = 105.42 \]
\[ \mathbb{E}(\hat{y}_{t+4}|x_{t+4}) = 0.9 \times 105.42 + 10 + 2 = 106.878 \]

... 

\[ \mathbb{E}(\hat{y}_{t=\infty}|x_{t=\infty}) = \frac{(10 + 2)}{(1 - 0.9)} = 120 \]

So is this a big effect or a small effect?
Interpreting AR(1) parameters

\[ \mathbb{E}(\hat{y}_t|x_t) = \frac{(10 + 2)}{(1 - 0.9)} = 120 \]

So is this a big effect or a small effect?

It depends on the length of time your covariates remain fixed.

Many social variables change rarely, so their effects accumulate slowly over time (e.g., institutions)

Presenting only \( \beta_1 \), rather than the accumulated change in \( y_t \) after \( x_t \) changes, could drastically *understate* the relative substantive importance of our social & political covariates compared to rapidly changing covariates

This understatement gets larger the closer \( \phi_1 \) gets to 1 — which is where our \( \phi_1 \)'s tend to be!

*A catch*: remember that if \( \phi_1 = 1 \), long-run predictions are impossible, so forecasting will produce misleading results of nonstationary processes
Interpreting AR(1) parameters

Recommendation:
Simulate the change in $y_t$ given a change in $x_t$ through enough periods to capture the real-world impact of your variables.

If you are studying partisan effects, and new parties tend to stay in power 5 years, don’t report $\beta_1$ or the one-year change in $y$. Iterate out to five years.

What is the confidence interval around these cumulative changes in $y$ given a permanent change in $x$?

A complex function of the se’s of $\phi$ and $\beta$

So simulate out to $y_{t+k}$ using draws from the estimated distributions of $\hat{\phi}$ and $\hat{\beta}$

R will help with this, using predict() and (in simcf), ldvsimev()
Example: UK vehicle accident deaths

Number of monthly deaths and serious injuries in UK road accidents

Data range from January 1969 to December 1984.

In February 1983, a new law requiring seat belt use took effect


http://www.staff.city.ac.uk/~sc397/courses/3ts/datasets.html

Simple, likely stationary data

Possibly seasonal

Simplest possible covariate: a single dummy
Vehicular accident deaths, UK, 1969–1984

The time series itself — looks cyclical, with a break in the series
The break corresponds closely with the change in seat belt laws

In a real data analysis, everything past this point is a bit gratuitous—this time series plot is simple and persuasive

But as most data analyses are more complex, this is a good testbed to learn techniques
What does this suggest?
What does this suggest?

How should we model seasons?
Decomposition of additive time series
November and December look especially dangerous

October and January look a bit dangerous

We could control for each month, select months, or Q4

This might also depend on serial correlation
## Model 1a: AR(1) specification

```r
## Estimate an AR(1) using arima
xcovariates <- law
arima.res1a <- arima(death, order = c(1,0,0),
                     xreg = xcovariates, include.mean = TRUE)
```

Coefficients:

<table>
<thead>
<tr>
<th></th>
<th>ar1</th>
<th>intercept</th>
<th>xcovariates</th>
</tr>
</thead>
<tbody>
<tr>
<td>coefficient</td>
<td>0.644</td>
<td>1719.19</td>
<td>-377.5</td>
</tr>
<tr>
<td>s.e.</td>
<td>0.055</td>
<td>42.08</td>
<td>107.7</td>
</tr>
</tbody>
</table>

\(\sigma^2\) estimated as 39289: log likelihood = -1288, aic = 2585

We begin with a simple model ignoring seasonality,
and controlling for one autoregressive lag
Model 1b: AR(1) specification with Q4 control

```r
## Estimate an AR(1) using arima
dxcovariates <- cbind(law, q4)
arima.res1b <- arima(death, order = c(1,0,0),
                        xreg = xcovariates, include.mean = TRUE)

Coefficients:

<table>
<thead>
<tr>
<th></th>
<th>ar1</th>
<th>intercept</th>
<th>law</th>
<th>q4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mean</td>
<td>0.5352</td>
<td>1638.0301</td>
<td>-395.6701</td>
<td>324.5653</td>
</tr>
<tr>
<td>S.E.</td>
<td>0.0636</td>
<td>28.1199</td>
<td>72.3030</td>
<td>34.5033</td>
</tr>
</tbody>
</table>

sigma^2 estimated as 26669: log likelihood = -1250.97, aic = 2511.93
```
Model 1c: AR(1) specification with all months

## Estimate an AR(1) using arima

\[
\text{xcovariates} \leftarrow \text{cbind(law, jan, feb, mar, apr, may, jun, aug, sep, oct, nov, dec)}
\]

\[
\text{arima.res1c} \leftarrow \text{arima(death, order = c(1,0,0),
} \\
\text{ \quad xreg = xcovariates, include.mean = TRUE)}
\]

Coefficients:

<table>
<thead>
<tr>
<th></th>
<th>ar1</th>
<th>intercept</th>
<th>law</th>
<th>jan</th>
<th>feb</th>
<th>mar</th>
<th>apr</th>
<th>may</th>
<th>jun</th>
<th>aug</th>
</tr>
</thead>
<tbody>
<tr>
<td>Estimate</td>
<td>0.6442</td>
<td>1638.6270</td>
<td>-370.0694</td>
<td>81.3021</td>
<td>-95.1350</td>
<td>-44.3298</td>
<td>-157.3445</td>
<td>-19.9428</td>
<td>-75.6674</td>
<td>14.7670</td>
</tr>
<tr>
<td>Std. Error</td>
<td>0.0550</td>
<td>42.9093</td>
<td>70.2727</td>
<td>54.8127</td>
<td>54.5036</td>
<td>53.0792</td>
<td>50.2149</td>
<td>45.0247</td>
<td>35.1890</td>
<td>35.1882</td>
</tr>
</tbody>
</table>

\[
sigma^2 \text{ estimated as 16333: log likelihood = -1204, aic = 2437.99}
\]
Model 1d: AR(1) specification with select months

## Estimate an AR(1) using arima

covariates <- cbind(law, jan, sep, oct, nov, dec)
arima.res1d <- arima(death, order = c(1,0,0),
                      xreg = covariates, include.mean = TRUE)

Coefficients:

<table>
<thead>
<tr>
<th></th>
<th>ar1</th>
<th>intercept</th>
<th>law</th>
<th>jan</th>
<th>sep</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0.6045</td>
<td>1589.4405</td>
<td>-377.7457</td>
<td>154.7288</td>
<td>80.7422</td>
</tr>
<tr>
<td>s.e.</td>
<td>0.0575</td>
<td>29.4161</td>
<td>69.7719</td>
<td>35.7336</td>
<td>35.8534</td>
</tr>
<tr>
<td></td>
<td>238.3880</td>
<td>451.3567</td>
<td>579.9770</td>
<td></td>
<td></td>
</tr>
<tr>
<td>s.e.</td>
<td>42.6836</td>
<td>44.3474</td>
<td>42.6108</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

sigma^2 estimated as 18989: log likelihood = -1218.42, aic = 2454.83
Model 1e: AR(1)AR(1)\_12 specification

```r
# Estimate an AR(1)AR(1)\_12 using arima
xcovariates <- cbind(law)
arima.res1e <- arima(death, order = c(1,0,0),
                     seasonal = list(order = c(1,0,0), period = 12),
                     xreg = xcovariates, include.mean = TRUE)

Coefficients:

<p>| | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>ar1</td>
<td>sar1</td>
<td>intercept</td>
<td>law</td>
<td></td>
</tr>
<tr>
<td>0.4446</td>
<td>0.6511</td>
<td>1710.1531</td>
<td>-347.6812</td>
<td></td>
</tr>
<tr>
<td>s.e.</td>
<td>0.0695</td>
<td>0.0564</td>
<td>53.3648</td>
<td>73.0634</td>
</tr>
</tbody>
</table>

sigma^2 estimated as 23693: log likelihood = -1242.86, aic = 2495.71
```
Model 1e: AR(1)AR(1)_{12} specification

Two questions:

1. Which model to select?
   Additive or multiplicative seasonality?
   A full set of month dummies, or a selection?

2. What is the effect of adding the law?
   In period $t + 1$, $t + 12$, $t + 60$

   How “significant” is this effect over those periods?
**Summary of fit so far**

<table>
<thead>
<tr>
<th>Model</th>
<th>Components</th>
<th>AIC</th>
<th>$\hat{\beta}_{\text{Law}}$</th>
<th>se($\hat{\beta}_{\text{Law}}$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1a</td>
<td>AR(1)</td>
<td>2585</td>
<td>-377</td>
<td>108</td>
</tr>
<tr>
<td>1b</td>
<td>AR(1), q4</td>
<td>2512</td>
<td>-396</td>
<td>72</td>
</tr>
<tr>
<td>1c</td>
<td>AR(1), all months</td>
<td>2438</td>
<td>-370</td>
<td>70</td>
</tr>
<tr>
<td>1d</td>
<td>AR(1), sep to jan</td>
<td>2455</td>
<td>-378</td>
<td>70</td>
</tr>
<tr>
<td>1e</td>
<td>AR(1)AR(1)$_{12}$</td>
<td>2496</td>
<td>-348</td>
<td>73</td>
</tr>
</tbody>
</table>

Which is the best fitting approach to seasonality?

Why did I use AIC to select models? What might be better?

What substantive difference does it make?

And what about higher order serial correlation?
An AR(p) Regression Model

The AR(p) regression model is a straightforward extension of the AR(1)

\[ y_t = y_{t-1}\phi_1 + y_{t-2}\phi_2 + \ldots + y_{t-p}\phi_p + x_t\beta + \varepsilon_t \]

Estimation is again by MLE, but similar to OLS with \( p \) lags of DV if \( t \) is large; MLE differs only in treatment of \( y_1 \) to \( y_p \)

Note that for fixed mean, \( y_t \) now converges to

\[ \mathbb{E}(y_t) = \frac{c}{1 - \phi_1 - \phi_2 - \phi_3 - \ldots - \phi_p} \]

Implication:
if, starting at time \( t \) and going forward to \( \infty \), we fix \( x_i\beta \),
then \( y_t \) will converge to \( x_i\beta/(1 - \phi_1 - \phi_2 - \phi_3 - \ldots - \phi_p) \)

Estimation and interpretation similar to above & uses same R functions
MA(1) Models

To create a regression model for an MA(1) process:

\[ y_t = \varepsilon_{t-1} \rho_1 + x_t \beta + \varepsilon_t \]

Estimation is again by maximum likelihood; no there is no obvious approximation to least squares.

Once again a complex procedure, but still a generalization of the Normal MLE.

Any dynamic effects in this model are quickly mean reverting.
ARMA(p,q): Putting it all together

To create a regression model for an ARMA(p,q) process:

\[ y_t = y_{t-1}\phi_1 + y_{t-2}\phi_2 + \ldots + y_{t-p}\phi_p + \varepsilon_{t-1}\rho_1 + \varepsilon_{t-2}\rho_2 + \ldots + \varepsilon_{t-q}\rho_q + x_t\beta + \varepsilon_t \]

We will need an MLE to obtain \( \hat{\phi}, \hat{\rho}, \) and \( \hat{\beta} \)

Once again a complex procedure, but still a generalization of the Normal case

Note the AR(p) process dominates in two senses:

- Stationarity determined just by AR(p) part of ARMA(p,q)
- Long-run level determined just by AR(p) terms: still \( x_i\beta/(1 - \sum_p \phi_p) \)
Model 2a: AR(2) specification

## Estimate an AR(2) using arima

```r
xcovariates <- cbind(law, jan, feb, mar, apr, may, jun, aug, sep, oct, 
                      nov, dec)

arima.res2a <- arima(death, order = c(2,0,0),
                      xreg = xcovariates, include.mean = TRUE)
```

Coefficients:

<table>
<thead>
<tr>
<th></th>
<th>ar1</th>
<th>ar2</th>
<th>intercept</th>
<th>law</th>
<th>jan</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0.4696</td>
<td>0.2711</td>
<td>1635.0869</td>
<td>-347.9213</td>
<td>83.7469</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>s.e.</th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0.0692</td>
<td>0.0694</td>
<td>45.6076</td>
<td>80.5683</td>
<td>46.9299</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>feb</th>
<th>mar</th>
<th>apr</th>
<th>may</th>
<th>jun</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>-94.9882</td>
<td>-44.0442</td>
<td>-157.2316</td>
<td>-19.8376</td>
<td>-75.5957</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>s.e.</th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>46.5145</td>
<td>45.0452</td>
<td>42.8448</td>
<td>37.9719</td>
<td>35.0631</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>aug</th>
<th>sep</th>
<th>oct</th>
<th>nov</th>
<th>dec</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>14.8059</td>
<td>67.5047</td>
<td>206.7362</td>
<td>406.0569</td>
<td>522.4596</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>s.e.</th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>35.0623</td>
<td>37.9640</td>
<td>42.8242</td>
<td>44.9760</td>
<td>46.4368</td>
</tr>
</tbody>
</table>

\[ \sigma^2 \text{ estimated as 15118: } \log \text{ likelihood} = -1196.65, \quad \text{aic} = 2425.3 \]
Model 2b: MA(1) specification

## Estimate an MA(1) using arima

xcovariates <- cbind(law, jan, feb, mar, apr, may, jun, aug, sep, oct, nov, dec)

arima.res2b <- arima(death, order = c(0,0,1),
           xreg = xcovariates, include.mean = TRUE
  )

Coefficients:

<table>
<thead>
<tr>
<th></th>
<th>ma1</th>
<th>intercept</th>
<th>law</th>
<th>jan</th>
<th>feb</th>
<th>mar</th>
<th>apr</th>
<th>may</th>
<th>jun</th>
<th>aug</th>
</tr>
</thead>
<tbody>
<tr>
<td>Coeff.</td>
<td>0.4539</td>
<td>1641.4834</td>
<td>-391.7280</td>
<td>79.9732</td>
<td>-94.6320</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>s.e.</td>
<td>0.0538</td>
<td>39.7814</td>
<td>45.5288</td>
<td>55.5797</td>
<td>55.6807</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Coeff.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>-44.0097</td>
<td>-157.2155</td>
<td>-19.8754</td>
<td>-75.6604</td>
<td>14.8400</td>
</tr>
<tr>
<td>s.e.</td>
<td>55.6807</td>
<td></td>
<td>55.6807</td>
<td>55.6807</td>
<td>43.9719</td>
<td>43.9719</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Coeff.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>s.e.</td>
<td>55.6807</td>
<td></td>
<td>55.6807</td>
<td>55.6807</td>
<td>55.6807</td>
<td>55.6807</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

sigma^2 estimated as 20566: log likelihood = -1225.97, aic = 2481.93
Model 2c: ARMA(1,1) specification

## Estimate an ARMA(1,1) using arima
xcovariates <- cbind(law, jan, feb, mar, apr, may, jun, aug, sep, oct, nov, dec)
arima.res2c <- arima(death, order = c(1,0,1),
xreg = xcovariates, include.mean = TRUE
)

Coefficients:

<table>
<thead>
<tr>
<th></th>
<th>ar1</th>
<th>ma1</th>
<th>intercept</th>
<th>law</th>
<th>jan</th>
</tr>
</thead>
<tbody>
<tr>
<td>Coeff</td>
<td>0.9349</td>
<td>-0.5994</td>
<td>1629.5549</td>
<td>-323.4929</td>
<td>85.7471</td>
</tr>
<tr>
<td>s.e.</td>
<td>0.0383</td>
<td>0.1076</td>
<td>58.6795</td>
<td>83.2081</td>
<td>40.4544</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>feb</th>
<th>mar</th>
<th>apr</th>
<th>may</th>
<th>jun</th>
</tr>
</thead>
<tbody>
<tr>
<td>Coeff</td>
<td>-94.0923</td>
<td>-43.6000</td>
<td>-156.8606</td>
<td>-19.6467</td>
<td>-75.5028</td>
</tr>
<tr>
<td>s.e.</td>
<td>40.2349</td>
<td>39.7247</td>
<td>38.8954</td>
<td>37.7225</td>
<td>36.1673</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>aug</th>
<th>sep</th>
<th>oct</th>
<th>nov</th>
<th>dec</th>
</tr>
</thead>
<tbody>
<tr>
<td>Coeff</td>
<td>14.7339</td>
<td>67.3872</td>
<td>206.5916</td>
<td>405.9572</td>
<td>522.3735</td>
</tr>
<tr>
<td>s.e.</td>
<td>36.1671</td>
<td>37.7207</td>
<td>38.8896</td>
<td>39.7111</td>
<td>40.2083</td>
</tr>
</tbody>
</table>

sigma^2 estimated as 14568: log likelihood = -1193.18, aic = 2418.37
Model 2d: ARMA(2,1) specification

```r
## Estimate an ARMA(2,1) using arima
xcovariates <- cbind(law, jan, feb, mar, apr, may, jun, aug, sep, oct, nov, dec)
arima.res2d <- arima(death, order = c(2,0,1),
                      xreg = xcovariates, include.mean = TRUE)

Coefficients:
                 ar1   ar2  ma1     intercept    law
  Coefficients:  1.1899 -0.2157 -0.7950 1626.1862 -321.2201
  s.e.  0.1071  0.0976  0.0724  68.6982  78.8301
                 jan   feb   mar    apr     may
                      84.8843 -94.5311 -43.8782 -157.0544 -19.7871
  s.e.  41.3869  41.3010  41.0435  40.5352  39.3222
                 jun    aug   sep    oct     nov
                      -75.5646  14.8208  67.5749 206.8634 406.3691
  s.e.  35.1484  35.1483  39.3216  40.5327  41.0341
                dec
                      522.9159
  s.e.  41.2487

sigma^2 estimated as 14284:  log likelihood = -1191.33,  aic = 2416.66
```
Model 2e: ARMA(1,2) specification

```r
## Estimate an ARMA(1,2) using arima
xcovariates <- cbind(law, jan, feb, mar, apr, may, jun, aug, sep, oct, nov, dec)
arima.res2e <- arima(death, order = c(1,0,2),
                     xreg = xcovariates, include.mean = TRUE)
```

Coefficients:

<table>
<thead>
<tr>
<th></th>
<th>ar1</th>
<th>ma1</th>
<th>ma2</th>
<th>intercept</th>
<th>law</th>
<th>jan</th>
<th>feb</th>
<th>mar</th>
<th>apr</th>
<th>may</th>
<th>jun</th>
<th>aug</th>
<th>sep</th>
<th>oct</th>
<th>nov</th>
<th>dec</th>
</tr>
</thead>
<tbody>
<tr>
<td>Estimate</td>
<td>0.9620</td>
<td>-0.5892</td>
<td>-0.1228</td>
<td>1627.146</td>
<td>-322.6854</td>
<td>85.1562</td>
<td>-94.1511</td>
<td>-43.6591</td>
<td>-156.9126</td>
<td>-19.6915</td>
<td>-75.5237</td>
<td>14.7645</td>
<td>67.4691</td>
<td>206.7084</td>
<td>406.1477</td>
<td>522.6613</td>
</tr>
<tr>
<td>s.e.</td>
<td>0.0253</td>
<td>0.0752</td>
<td>0.0705</td>
<td>66.814</td>
<td>79.2449</td>
<td>40.7504</td>
<td>40.6400</td>
<td>40.3701</td>
<td>39.9498</td>
<td>39.3736</td>
<td>35.5453</td>
<td>35.5453</td>
<td>39.3730</td>
<td>39.9476</td>
<td>40.3650</td>
<td>40.5994</td>
</tr>
</tbody>
</table>

sigma^2 estimated as 14356: log likelihood = -1191.82, aic = 2417.63
Model 2f: ARMA(2,2) specification

## Estimate an ARMA(2,2) using arima

xcovariates <- cbind(law, jan, feb, mar, apr, may, jun, aug, sep, oct, 
                    nov, dec)
arima.res2f <- arima(death, order = c(2,0,2), 
                     xreg = xcovariates, include.mean = TRUE)

Coefficients:

<table>
<thead>
<tr>
<th></th>
<th>ar1</th>
<th>ar2</th>
<th>ma1</th>
<th>ma2</th>
<th>intercept</th>
</tr>
</thead>
<tbody>
<tr>
<td>Coefs.</td>
<td>0.0526</td>
<td>0.8449</td>
<td>0.3497</td>
<td>-0.6503</td>
<td>1625.7793</td>
</tr>
<tr>
<td>s.e.</td>
<td>0.0538</td>
<td>0.0413</td>
<td>0.1006</td>
<td>0.0998</td>
<td>61.5565</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>law</th>
<th>jan</th>
<th>feb</th>
<th>mar</th>
<th>apr</th>
</tr>
</thead>
<tbody>
<tr>
<td>Coefs.</td>
<td>-312.2308</td>
<td>86.0931</td>
<td>-91.7482</td>
<td>-43.7677</td>
<td>-154.3960</td>
</tr>
<tr>
<td>s.e.</td>
<td>81.8335</td>
<td>40.9421</td>
<td>38.1258</td>
<td>40.4084</td>
<td>36.9053</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>may</th>
<th>jun</th>
<th>aug</th>
<th>sep</th>
<th>oct</th>
</tr>
</thead>
<tbody>
<tr>
<td>Coefs.</td>
<td>-19.6984</td>
<td>-72.8430</td>
<td>17.6629</td>
<td>67.3856</td>
<td>209.8757</td>
</tr>
<tr>
<td>s.e.</td>
<td>38.9443</td>
<td>34.4385</td>
<td>34.4299</td>
<td>38.9431</td>
<td>36.8765</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>nov</th>
<th>dec</th>
</tr>
</thead>
<tbody>
<tr>
<td>Coefs.</td>
<td>405.8869</td>
<td>526.1152</td>
</tr>
<tr>
<td>s.e.</td>
<td>40.3991</td>
<td>38.0647</td>
</tr>
</tbody>
</table>

sigma^2 estimated as 13794: log likelihood = -1189.2, aic = 2414.39
Whew!

This gets tedious fast. . .

To have R search automatically for a low AIC model, try `auto.arima()` in the `forecast` library.

This gets complicated if the series is potentially seasonal and/or nonstationary.

My practice: search/diagnose manually where feasible, automatically where many runs are needed (e.g., 1 million time series analyses?)

More on this in lab. . .
## Summary of fit

<table>
<thead>
<tr>
<th>Model</th>
<th>Components</th>
<th>AIC</th>
<th>$\hat{\beta}_{\text{Law}}$</th>
<th>$\text{se}(\hat{\beta}_{\text{Law}})$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1a</td>
<td>AR(1)</td>
<td>2585</td>
<td>$-377$</td>
<td>108</td>
</tr>
<tr>
<td>1b</td>
<td>AR(1), q4</td>
<td>2512</td>
<td>$-396$</td>
<td>72</td>
</tr>
<tr>
<td>1c</td>
<td>AR(1), all months</td>
<td>2438</td>
<td>$-370$</td>
<td>70</td>
</tr>
<tr>
<td>1d</td>
<td>AR(1), sep to jan</td>
<td>2455</td>
<td>$-378$</td>
<td>70</td>
</tr>
<tr>
<td>1e</td>
<td>AR(1)AR(1)$_{12}$</td>
<td>2496</td>
<td>$-348$</td>
<td>73</td>
</tr>
<tr>
<td>2a</td>
<td>AR(2), all months</td>
<td>2425</td>
<td>$-348$</td>
<td>81</td>
</tr>
<tr>
<td>2b</td>
<td>MA(1), all months</td>
<td>2482</td>
<td>$-392$</td>
<td>46</td>
</tr>
<tr>
<td>2c</td>
<td>ARMA(1,1), all months</td>
<td>2418</td>
<td>$-323$</td>
<td>83</td>
</tr>
<tr>
<td>2d,3a</td>
<td>ARMA(2,1), all months</td>
<td>2417</td>
<td>$-321$</td>
<td>79</td>
</tr>
<tr>
<td>2e</td>
<td>ARMA(1,2), all months</td>
<td>2418</td>
<td>$-323$</td>
<td>79</td>
</tr>
<tr>
<td>2f</td>
<td>ARMA(2,2), all months</td>
<td>2414</td>
<td>$-321$</td>
<td>79</td>
</tr>
</tbody>
</table>

Which model looks best?

What might be a better way to judge than AIC?
Cross-validation

Out of sample tests of fit are more reliable than in sample tests

But what is out-of-sample in time series?

Can’t just pull random observations out of sequence:
best CV method for time series is a rolling forecast window

Issue for all cross-validation:
danger of collinearity if you have binary covariates that change rarely!
### Summary of fit

<table>
<thead>
<tr>
<th>Model</th>
<th>Components</th>
<th>AIC</th>
<th>cv1-MAE</th>
<th>β_{Law}</th>
<th>se(β_{Law})</th>
</tr>
</thead>
<tbody>
<tr>
<td>1a</td>
<td>AR(1)</td>
<td>2585</td>
<td>120.4</td>
<td>-377</td>
<td>108</td>
</tr>
<tr>
<td>1b</td>
<td>AR(1), q4</td>
<td>2512</td>
<td>108.5</td>
<td>-396</td>
<td>72</td>
</tr>
<tr>
<td>1c</td>
<td>AR(1), all months</td>
<td>2438</td>
<td>83.9</td>
<td>-370</td>
<td>70</td>
</tr>
<tr>
<td>1d</td>
<td>AR(1), sep to jan</td>
<td>2455</td>
<td>119.7</td>
<td>-378</td>
<td>70</td>
</tr>
<tr>
<td>1e</td>
<td>AR(1)AR(1)_{12}</td>
<td>2496</td>
<td>119.7</td>
<td>-348</td>
<td>73</td>
</tr>
<tr>
<td>2a</td>
<td>AR(2), all months</td>
<td>2425</td>
<td>92.6</td>
<td>-348</td>
<td>81</td>
</tr>
<tr>
<td>2b</td>
<td>MA(1), all months</td>
<td>2482</td>
<td>79.9</td>
<td>-392</td>
<td>46</td>
</tr>
<tr>
<td>2c</td>
<td>ARMA(1,1), all months</td>
<td>2418</td>
<td>89.5</td>
<td>-323</td>
<td>83</td>
</tr>
<tr>
<td>2d,3a</td>
<td>ARMA(2,1), all months</td>
<td>2417</td>
<td>83.5</td>
<td>-321</td>
<td>79</td>
</tr>
<tr>
<td>2e</td>
<td>ARMA(1,2), all months</td>
<td>2418</td>
<td>84.8</td>
<td>-323</td>
<td>79</td>
</tr>
<tr>
<td>2f</td>
<td>ARMA(2,2), all months</td>
<td>2414</td>
<td>85.5</td>
<td>-321</td>
<td>79</td>
</tr>
</tbody>
</table>

We could look at the one-period-ahead out-of-sample forecast

cv1-MAE shows the mean absolute error in this prediction

Note we have fairly few periods left to forecast, as we need a long window to estimate the effect of the law (which doesn't start until period 170)
<table>
<thead>
<tr>
<th>Model</th>
<th>Components</th>
<th>AIC</th>
<th>cv1-MAE</th>
<th>$\beta_{\text{Law}}$</th>
<th>se($\beta_{\text{Law}}$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1a</td>
<td>AR(1)</td>
<td>2585</td>
<td>120.4</td>
<td>-377</td>
<td>108</td>
</tr>
<tr>
<td>1b</td>
<td>AR(1), q4</td>
<td>2512</td>
<td>108.5</td>
<td>-396</td>
<td>72</td>
</tr>
<tr>
<td>1c</td>
<td>AR(1), all months</td>
<td>2438</td>
<td>83.9</td>
<td>-370</td>
<td>70</td>
</tr>
<tr>
<td>1d</td>
<td>AR(1), sep to jan</td>
<td>2455</td>
<td>119.7</td>
<td>-378</td>
<td>70</td>
</tr>
<tr>
<td>1e</td>
<td>AR(1)AR(1)$_{12}$</td>
<td>2496</td>
<td>119.7</td>
<td>-348</td>
<td>73</td>
</tr>
<tr>
<td>2a</td>
<td>AR(2), all months</td>
<td>2425</td>
<td>92.6</td>
<td>-348</td>
<td>81</td>
</tr>
<tr>
<td>2b</td>
<td>MA(1), all months</td>
<td>2482</td>
<td>79.9</td>
<td>-392</td>
<td>46</td>
</tr>
<tr>
<td>2c</td>
<td>ARMA(1,1), all months</td>
<td>2418</td>
<td>89.5</td>
<td>-323</td>
<td>83</td>
</tr>
<tr>
<td>2d,3a</td>
<td>ARMA(2,1), all months</td>
<td>2417</td>
<td>83.5</td>
<td>-321</td>
<td>79</td>
</tr>
<tr>
<td>2e</td>
<td>ARMA(1,2), all months</td>
<td>2418</td>
<td>84.8</td>
<td>-323</td>
<td>79</td>
</tr>
<tr>
<td>2f</td>
<td>ARMA(2,2), all months</td>
<td>2414</td>
<td>85.5</td>
<td>-321</td>
<td>79</td>
</tr>
</tbody>
</table>

Is MA(1) really the best fitting model, against the in-sample evidence?

Perhaps we should look at forecasts beyond one period ahead?

A graphic helps...
Some models seem easy to reject, but which is/are best?
Do we trust the predictions out at 10–12 months? Why or why not?
## Summary of fit

<table>
<thead>
<tr>
<th>Model</th>
<th>Components</th>
<th>AIC</th>
<th>cv12-MAE</th>
<th>$\beta_{\text{Law}}$</th>
<th>se($\beta_{\text{Law}}$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1a</td>
<td>AR(1)</td>
<td>2585</td>
<td>166.4</td>
<td>−377</td>
<td>108</td>
</tr>
<tr>
<td>1b</td>
<td>AR(1), q4</td>
<td>2512</td>
<td>101.5</td>
<td>−396</td>
<td>72</td>
</tr>
<tr>
<td>1c</td>
<td>AR(1), all months</td>
<td>2438</td>
<td>80.4</td>
<td>−370</td>
<td>70</td>
</tr>
<tr>
<td>1d</td>
<td>AR(1), sep to jan</td>
<td>2455</td>
<td>93.6</td>
<td>−378</td>
<td>70</td>
</tr>
<tr>
<td>1e</td>
<td>AR(1)AR(1)$_{12}$</td>
<td>2496</td>
<td>170.7</td>
<td>−348</td>
<td>73</td>
</tr>
<tr>
<td>2a</td>
<td>AR(2), all months</td>
<td>2425</td>
<td>81.4</td>
<td>−348</td>
<td>81</td>
</tr>
<tr>
<td>2b</td>
<td>MA(1), all months</td>
<td>2482</td>
<td>83.7</td>
<td>−392</td>
<td>46</td>
</tr>
<tr>
<td>2c</td>
<td>ARMA(1,1), all months</td>
<td>2418</td>
<td>79.5</td>
<td>−323</td>
<td>83</td>
</tr>
<tr>
<td>2d,3a</td>
<td>ARMA(2,1), all months</td>
<td>2417</td>
<td>80.3</td>
<td>−321</td>
<td>79</td>
</tr>
<tr>
<td>2e</td>
<td>ARMA(1,2), all months</td>
<td>2418</td>
<td>80.1</td>
<td>−323</td>
<td>79</td>
</tr>
<tr>
<td>2f</td>
<td>ARMA(2,2), all months</td>
<td>2414</td>
<td>78.1</td>
<td>−321</td>
<td>79</td>
</tr>
</tbody>
</table>

We might summarize the prior figure with the average MAE averaged over the 12 month forecast.

This suggests similar performance for most ARMA models, except MA(1), which is poorer.
### Summary of fit

<table>
<thead>
<tr>
<th>Model</th>
<th>Components</th>
<th>AIC</th>
<th>cv8-MAE</th>
<th>$\beta_{\text{Law}}$</th>
<th>$\text{se}(\beta_{\text{Law}})$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1a</td>
<td>AR(1)</td>
<td>2585</td>
<td>168.0</td>
<td>-377</td>
<td>108</td>
</tr>
<tr>
<td>1b</td>
<td>AR(1), q4</td>
<td>2512</td>
<td>96.2</td>
<td>-396</td>
<td>72</td>
</tr>
<tr>
<td>1c</td>
<td>AR(1), all months</td>
<td>2438</td>
<td>87.2</td>
<td>-370</td>
<td>70</td>
</tr>
<tr>
<td>1d</td>
<td>AR(1), sep to jan</td>
<td>2455</td>
<td>97.9</td>
<td>-378</td>
<td>70</td>
</tr>
<tr>
<td>1e</td>
<td>AR(1)AR(1)$_{12}$</td>
<td>2496</td>
<td>171.7</td>
<td>-348</td>
<td>73</td>
</tr>
<tr>
<td>2a</td>
<td>AR(2), all months</td>
<td>2425</td>
<td>89.5</td>
<td>-348</td>
<td>81</td>
</tr>
<tr>
<td>2b</td>
<td>MA(1), all months</td>
<td>2482</td>
<td>89.1</td>
<td>-392</td>
<td>46</td>
</tr>
<tr>
<td>2c</td>
<td>ARMA(1,1), all months</td>
<td>2418</td>
<td>87.2</td>
<td>-323</td>
<td>83</td>
</tr>
<tr>
<td>2d,3a</td>
<td>ARMA(2,1), all months</td>
<td>2417</td>
<td>87.0</td>
<td>-321</td>
<td>79</td>
</tr>
<tr>
<td>2e</td>
<td>ARMA(1,2), all months</td>
<td>2418</td>
<td>87.0</td>
<td>-323</td>
<td>79</td>
</tr>
<tr>
<td>2f</td>
<td>ARMA(2,2), all months</td>
<td>2414</td>
<td>86.0</td>
<td>-321</td>
<td>79</td>
</tr>
</tbody>
</table>

Even discounting the forecasts past 8 months, model 2f comes out slightly ahead. . .

But all of the models with monthly controls and at least one AR term do roughly equally well.
Selected model: ARMA(2,2)

Coefficients:

<table>
<thead>
<tr>
<th></th>
<th>ar1</th>
<th>ar2</th>
<th>ma1</th>
<th>ma2</th>
<th>intercept</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0.0526</td>
<td>0.8449</td>
<td>0.3497</td>
<td>-0.6503</td>
<td>1625.7793</td>
</tr>
<tr>
<td>s.e.</td>
<td>0.0538</td>
<td>0.0413</td>
<td>0.1006</td>
<td>0.0998</td>
<td>61.5565</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>law</th>
<th>jan</th>
<th>feb</th>
<th>mar</th>
<th>apr</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>-312.2308</td>
<td>86.0931</td>
<td>-91.7482</td>
<td>-43.7677</td>
<td>-154.3960</td>
</tr>
<tr>
<td>s.e.</td>
<td>81.8335</td>
<td>40.9421</td>
<td>38.1258</td>
<td>40.4084</td>
<td>36.9053</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>may</th>
<th>jun</th>
<th>aug</th>
<th>sep</th>
<th>oct</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>-19.6984</td>
<td>-72.8430</td>
<td>17.6629</td>
<td>67.3856</td>
<td>209.8757</td>
</tr>
<tr>
<td>s.e.</td>
<td>38.9443</td>
<td>34.4385</td>
<td>34.4299</td>
<td>38.9431</td>
<td>36.8765</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>nov</th>
<th>dec</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>405.8869</td>
<td>526.1152</td>
</tr>
<tr>
<td>s.e.</td>
<td>40.3991</td>
<td>38.0647</td>
</tr>
</tbody>
</table>

\[ \sigma^2 \text{ estimated as } 13794: \log \text{ likelihood} = -1189.2, \ aic = 2414.39 \]

We have a model – but what does it mean?

Where does this series go over time, with or without the law?
Counterfactual forecasting

We consider two algorithms for forecasting:

Both assume we have point estimates and the variance covariance matrix of the model parameters, $\hat{\beta}$, $\hat{\phi}$, $\hat{\rho}$

Both compute forecast over the next $K$ periods given hypothetical values of the covariates, $x_{c,t+1}, \ldots, x_{c,t+k}$

Both forecasts are uncertain due to uncertainty in model parameter estimates

Approach 1: **predicted values** $\tilde{y}_{t+k}$, which include the uncertainty due to shocks, $\varepsilon_{t+1}, \ldots, \varepsilon_{t+K}$

For this approach, we also need the estimated variance of these shocks, $\hat{\sigma}^2$

Approach 2: **expected values** $\hat{y}_{t+k}$, which average over the anticipated shocks

Expected values show the expected path of the outcome over the next $K$ periods, given the counterfactual covariates
Counterfactual forecasting: Predicted Values

1. Start in period $t$ with the observed $y_t$ and $x_t$; choose hypothetical $x_{c,t+k}$’s for each period $t + 1, \ldots, t + k, \ldots, t + K$ forecast.

2. Draw a vector of simulated parameters from their asymptotic distribution:

\[
\text{vec}\left(\tilde{\beta}, \tilde{\phi}, \tilde{\rho}\right) \sim \mathcal{MVN}\left(\text{vec}\left(\hat{\beta}, \hat{\phi}, \hat{\rho}\right), \text{Var}\left(\text{vec}\left(\hat{\beta}, \hat{\phi}, \hat{\rho}\right)\right)\right).
\]

3. Iterate over the following steps for forecast period $k$ in $1, \ldots, K$:

   (a) Draw a new random shock $\tilde{\varepsilon}_{t+1} \sim \mathcal{N}(0, \hat{\sigma}^2)$.
   (b) Calculate one simulated predicted value, $\tilde{y}_{t+k}$ using

\[
\tilde{y}_{t+k} = \sum_{p=1}^{P} y_{t+k-p}\tilde{\phi}_p + x_{c,t+k}\tilde{\beta} + \sum_{q=1}^{Q} \tilde{\varepsilon}_{t+k-q}\tilde{\rho}_q + \tilde{\varepsilon}_{t+k}.
\]

This formula uses past values of $y$ and $\varepsilon$, which may be simulates from prior iterations of the forecast.

4. Repeat steps 2 and 3 $\text{sim}\times$ times to construct $\text{sim}$ simulated forecasts. Summarize these predicted values by means and quantiles (predictive intervals).
Counterfactual forecasting: Expected Values

1. Start in period $t$ with the observed $y_t$ and $x_t$; choose hypothetical $x_{c,t+k}$’s for each period $t + 1, \ldots, t + k, \ldots, t + K$ forecast.

2. Draw a vector of simulated parameters from their asymptotic distribution:
   \[ \text{vec} \left( \tilde{\beta}, \tilde{\phi}, \tilde{\rho} \right) \sim \mathcal{MVN} \left( \text{vec} \left( \hat{\beta}, \hat{\phi}, \hat{\rho} \right), \text{Var} \left( \text{vec} \left( \hat{\beta}, \hat{\phi}, \hat{\rho} \right) \right) \right) \]

3. Iterate over the following step for forecast period $k$ in $1, \ldots, K$:
   
   (a) Calculate one simulated expected value of $y_{t+k}$ using
   \[ \mathbb{E} \left( \tilde{y}_{t+k} \mid \tilde{\beta}, \tilde{\phi}, \tilde{\rho}, x_{c,t}, \ldots, x_{c,t+k}, y_t \right) = \sum_{p=1}^{P} y_{t+k-p} \tilde{\phi}_p + x_{c,t+k} \tilde{\beta}. \]
   
   This formula uses past values of $y$ and $\varepsilon$, which may be simulates from prior iterations of the forecast.

4. Repeat steps 2 and 3 $\text{sims}$ times to construct $\text{sims}$ simulated forecasts.
   Summarize these expected values by means and quantiles (confidence intervals).
Effect of repealing seatbelt law?

What does the model predict would happen if we repealed the law?

How much would deaths increase after one month? One year? Five years?

If we run this experiment, how much might the results vary from model expectations?

Need forecast deaths—no law for the next 60 periods, plus predictive intervals

```r
predict(arima.res1,  # The model
    n.ahead = 60,  # predict out 60 periods
    newxreg = newdata)  # using these counterfactual x’s
```
The observed time series
What the model predicts would happen if the seat belt requirement is *repealed*
Predicted effect of reversing seat belt law

adding the 95% predictive interval
Predicted effect of reversing seat belt law

which is easier to read as a polygon
Predicted effect of reversing seat belt law

comparing to what would happen with the law left intact
Predicted effect of reversing seat belt law

comparing to what would happen with the law left intact
Confidence intervals vs. Predictive Intervals

Suppose we want *confidence intervals* instead of *predictive intervals*

CIs just show the uncertainty from estimation

Analog to $\text{se}(\beta)$ and significance tests

`predict.arima()` won’t give us CIs

Need to use another package, or `simcf` (later in the course)
Neat. But is ARMA\((p,q)\) appropriate for our data?

ARMA\((p,q)\) an extremely flexible, broadly applicable model of single time series \(y_t\)

But ONLY IF \(y_t\) is stationary

If data are non-stationary (have a unit root), then:

- Results may be spurrious
- Long-run predictions impossible

Can assess stationarity through two methods:

1. Examine the data: time series, ACF, and PACF plots
2. Statistical tests for a unit root
Unit root tests: Basic notion

- If $y_t$ is stationary, large negative shifts should be followed by large positive shifts, and vice versa (mean-reversion)

- If $y_t$ is non-stationary (has a unit root), large negative shifts should be uncorrelated with large positive shifts

Thus if we regress $y_t - y_{t-1}$ on $y_{t-1}$, we should get a negative coefficient if and only if the series is stationary

To do this:

Augmented Dickey-Fuller test `adf.test()` in the `tseries` library

Phillips-Perron test: `PP.test()`

Tests differ in how they model heteroskedasticity, serial correlation, and the number of lags
Unit root tests: Limitations

Form of unit root test: rejecting the null of a unit root

Will tend to fail to reject for many non-unit roots with high persistence

Very hard to distinguish near-unit roots from unit roots with test statistics

Famously low power tests for single time series
Unit root tests: Limitations

Analogy: Using polling data to predict a very close election

Null Hypothesis: Left Party will get 50.01% of the vote

Alternative Hypothesis: Left will get < 50% of the vote

We’re okay with a 3% CI if we’re interested in alternatives like 45% of the vote

But suppose we need to compare the Null to 49.99%

To confidently reject the Null in favor of a very close alternative like this, we’d need a CI of about 0.005% or less
Unit root tests: Limitations

In many political science applications, we ask whether $\phi = 1$ or, say, $\phi = 0.95$.

Small numerical difference makes a huge difference for modeling.

And single-series unit root tests are weak, and poorly discriminate across these cases.

Simply not much use to us for a single time series, unless we have panel data.

Then we can use panel versions of unit root tests that have somewhat more power.

More about panel unit root tests later in the course.
Unit root tests: usage

> # Check for a unit root
> PP.test(death)

Phillips-Perron Unit Root Test

data: death
Dickey-Fuller = -6.435, Truncation lag parameter = 4, p-value = 0.01

> adf.test(death)

Augmented Dickey-Fuller Test

data: death
Dickey-Fuller = -6.537, Lag order = 5, p-value = 0.01
alternative hypothesis: stationary
Linear regression with $y_{t-1}$ as a control

A popular model in comparative politics & political science is:

$$y_t = y_{t-1} \phi_1 + x_t \beta + \epsilon_t$$

estimated by least squares, rather than maximum likelihood

That is, treat $y_{t-1}$ as “just another covariate”, rather than a special term

Danger of this approach: $y_{t-1}$ and $\epsilon_t$ are almost certainly correlated (Why?)

Unless we model serial correlation correctly, our errors will be serially correlated, and last period’s error is definitely correlated with last period’s realization

So if $y_{t-1}$ is treated as a covariate in a linear regression, this violates G-M condition 2, which requires that $\mathbb{E}(x_i \epsilon_i) = 0$

The consequences could be bias in $\hat{\beta}$ and incorrect s.e.’s
When can you use a lag of $y$ as a control in OLS?

My recommendation:

1. Estimate an LS model with the lagged DV
2. Check for remaining serial correlation (Breusch-Godfrey)
3. Compare your results to the corresponding AR(p) estimated by MLE
4. Consider ARMA(p,q) alternatives estimated by MLE
5. Use LS only if it make no statistical or substantive difference

Upshot: You can use LS in cases where it works just as well as MLE

If you model the right number of lags, and need no MA(q) terms, and have lots of time periods, LS often not far off

Be skeptical of LS standard errors that disagree with AR(p)

Still need to interpret the $\beta$’s and $\phi$’s dynamically
Testing for serial correlation in errors

In LS models, serial correlation makes estimates inefficient (like heteroskedasticity)

*If* the model includes a lagged dependent variable, serial correlation $\rightarrow$ inconsistent estimates: $\mathbb{E}(x\varepsilon) \neq 0$

So we need to be able to test for serial correlation.

A general test that will work for single time series or panel data is based on the Lagrange Multiplier

Called Breusch-Godfrey test, or the LM test
Lagrange Multiplier test for serial correlation

1. Run your time series regression by least squares, regressing

\[ y_t = \beta_0 + \beta_1 x_{1t} + \ldots + \beta_k x_{kt} + \phi_1 y_{t-1} + \ldots + \phi_k y_{t-p} + u_t \]

2. Regress (using LS) \( \hat{u}_t \) on a constant, the explanatory variables \( x_1, \ldots, x_k, y_{t-1}, \ldots, y_{t-m} \), and the lagged residuals, \( \hat{u}_{t-1}, \ldots \hat{u}_{t-m} \)

Be sure to chose \( m \leq p \). If you choose \( m = 1 \), you have a test for 1st degree autocorrelation; if you choose \( m = 2 \), you have a test for 2nd degree autocorrelation, etc.

3. Compute the test-statistic \( (T-m)R^2 \), where \( R^2 \) is the coefficient of determination from the regression in step 2. This test statistic is distributed \( \chi^2 \) with \( m \) degrees of freedom.

4. Rejecting the null for this test statistic is equivalent to rejecting no autocorrelation.
Regression with lagged DV for Accidents

Call:
\[ \text{lm(formula = death ~ lagdeath + jan + feb + mar + apr + may + jun + aug + sep + oct + nov + dec + law)} \]

Residuals:

<table>
<thead>
<tr>
<th></th>
<th>Min</th>
<th>1Q</th>
<th>Median</th>
<th>3Q</th>
<th>Max</th>
</tr>
</thead>
<tbody>
<tr>
<td>Min</td>
<td>-323.58</td>
<td>-84.45</td>
<td>-3.80</td>
<td>80.97</td>
<td>404.88</td>
</tr>
</tbody>
</table>

Coefficients:

|                  | Estimate | Std. Error | t value | Pr(>|t|) |
|------------------|----------|------------|---------|---------|
| (Intercept)      | 635.11393| 96.64706   | 6.571   | 5.38e-10 *** |
| lagdeath         | 0.64313 | 0.05787    | 11.114  | < 2e-16 *** |
| jan              | -302.58936 | 59.33982 | -5.099  | 8.71e-07 *** |
| feb              | -211.00947 | 48.46926 | -4.353  | 2.26e-05 *** |
| mar              | -31.82070 | 47.33602 | -0.672  | 0.502314 |
| apr              | -177.52653 | 47.35870 | -3.749  | 0.000241 *** |
| may              | 32.58040 | 47.55810 | 0.685   | 0.494199 |
| jun              | -111.47957 | 47.43316 | -2.350  | 0.019863 * |
| aug              | -33.76181 | 47.52523 | -0.710  | 0.478393 |
| sep              | 9.48411 | 47.61220 | 0.199   | 0.842339 |
| oct              | 114.89374 | 48.04444 | 2.391   | 0.017832 * |
nov   224.81981   50.07068   4.490  1.28e-05    ***  
dec   213.09991   54.93824   3.879  0.000148    ***  
law  -145.31036   37.36477  -3.889  0.000142    ***  
---
Signif. codes:  
        0 ***  0.001 **  0.01 *  0.05 .  0.1  1

Residual standard error: 133.9 on 177 degrees of freedom  
(1 observation deleted due to missingness)  
Multiple R-squared:  0.802, Adjusted R-squared:  0.7875  
F-statistic: 55.17 on 13 and 177 DF,  p-value: < 2.2e-16
Tests for serial correlation

Breusch-Godfrey test for serial correlation of order up to 1

data:  lm.res1f
LM test = 11.5457, df = 1, p-value = 0.000679

Breusch-Godfrey test for serial correlation of order up to 2

data:  lm.res1f
LM test = 11.9843, df = 2, p-value = 0.002498

Clear evidence of residual serial correlation
Regression with two lags of DV for Accidents

Call:
\text{lm}(\text{formula} = \text{death} \sim \text{lagdeath} + \text{lag2death} + \text{jan} + \text{feb} + \text{mar} + \\
\text{apr} + \text{may} + \text{jun} + \text{aug} + \text{sep} + \text{oct} + \text{nov} + \text{dec} + \text{law})

Residuals:

<table>
<thead>
<tr>
<th></th>
<th>Min</th>
<th>1Q</th>
<th>Median</th>
<th>3Q</th>
<th>Max</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>-378.22</td>
<td>-88.29</td>
<td>-5.04</td>
<td>89.71</td>
<td>308.44</td>
</tr>
</tbody>
</table>

Coefficients:

|                    | Estimate | Std. Error | t value | Pr(>|t|) |
|--------------------|----------|------------|---------|----------|
| (Intercept)        | 475.12645| 103.68324  | 4.582   | 8.71e-06 *** |
| lagdeath           | 0.47250  | 0.07332    | 6.445   | 1.09e-09 *** |
| lag2death          | 0.26362  | 0.07284    | 3.619   | 0.000387 *** |
| jan                | -311.45937| 57.62112   | -5.405  | 2.09e-07 *** |
| feb                | -329.58156| 57.96856   | -5.686  | 5.37e-08 *** |
| mar                | -68.08737| 46.99905   | -1.449  | 0.149212 |
| apr                | -152.44095| 46.46031   | -3.281  | 0.001248 ** |
| may                | 25.02334 | 46.18114   | 0.542   | 0.588610 |
| jun                | -65.76811| 47.71466   | -1.378  | 0.169851 |
| aug                | -6.16090 | 46.72852   | -0.132  | 0.895259 |
| sep                | 19.68658 | 46.27238   | 0.425   | 0.671032 |
oct    130.18618  46.79714  2.782  0.005997  **
nov    249.97112  49.06743  5.094  9.00e-07  ***
dec    235.55993  53.65766  4.390  1.96e-05  ***
law   -111.47166  37.45979  -2.976  0.003336  **

Signif. codes:
0 *** 0.001 ** 0.01 * 0.05 . 0.1  1

Residual standard error: 129.8 on 175 degrees of freedom
(2 observations deleted due to missingness)
Multiple R-squared:  0.8155, Adjusted R-squared:  0.8008
F-statistic: 55.26 on 14 and 175 DF,  p-value: < 2.2e-16
Tests for serial correlation

Breusch-Godfrey test for serial correlation of order up to 1

data:  lm.res1g
LM test = 0.6961, df = 1, p-value = 0.4041

> bgtest(lm.res1g,2)

Breusch-Godfrey test for serial correlation of order up to 2

data:  lm.res1g
LM test = 3.2256, df = 2, p-value = 0.1993

Perhaps some weak evidence of residual serial correlation, but as with other tests, hard to be sure if we need to go beyond AR(2)