MONASH University

MONASH BUSINESS SCHOOL

# Forecasting: principles and practice

Rob J Hyndman

## 3.2 Dynamic regression

# Outline

# Regression with ARIMA errors

## Regression models

$$y_t = \beta_0 + \beta_1 x_{1,t} + \cdots + \beta_k x_{k,t} + e_t,$$

- $y_t$ modeled as function of $k$ explanatory variables $x_{1,t}, \dots, x_{k,t}$.
- In regression, we assume that $e_t$ was WN.
- Now we want to allow $e_t$ to be autocorrelated.

## Example: ARIMA(1,1,1) errors

$$y_t = \beta_0 + \beta_1 x_{1,t} + \cdots + \beta_k x_{k,t} + n_t,$$
$$(1 - \phi_1 B)(1 - B)n_t = (1 + \theta_1 B)e_t,$$

where $e_t$ is white noise .

# Regression with ARIMA errors

## Regression models

$$y_t = \beta_0 + \beta_1 x_{1,t} + \cdots + \beta_k x_{k,t} + e_t,$$

- $y_t$ modeled as function of $k$ explanatory variables $x_{1,t}, \ldots, x_{k,t}$.
- In regression, we assume that $e_t$ was WN.
- Now we want to allow $e_t$ to be autocorrelated.

## Example: ARIMA(1,1,1) errors

$$y_t = \beta_0 + \beta_1 x_{1,t} + \cdots + \beta_k x_{k,t} + n_t,$$
$$(1 - \phi_1 B)(1 - B) n_t = (1 + \theta_1 B) e_t,$$

where $e_t$ is white noise .

# Residuals and errors

**Example: $n_t$ = ARIMA(1,1,1)**

$$y_t = \beta_0 + \beta_1 x_{1,t} + \cdots + \beta_k x_{k,t} + n_t,$$
$$(1 - \phi_1 B)(1 - B)n_t = (1 + \theta_1 B)e_t,$$

- Be careful in distinguishing $n_t$ from $e_t$.
- Only the errors $n_t$ are assumed to be white noise.
- In ordinary regression, $n_t$ is assumed to be white noise and so $n_t = e_t$.

# Residuals and errors

**Example: $n_t$ = ARIMA(1,1,1)**

$$y_t = \beta_0 + \beta_1 x_{1,t} + \cdots + \beta_k x_{k,t} + n_t,$$
$$(1 - \phi_1 B)(1 - B)n_t = (1 + \theta_1 B)e_t,$$

- Be careful in distinguishing $n_t$ from $e_t$.
- Only the errors $n_t$ are assumed to be white noise.
- In ordinary regression, $n_t$ is assumed to be white noise and so $n_t = e_t$.

# Estimation

If we minimize $\sum n_t^2$ (by using ordinary regression):

1. Estimated coefficients $\hat{\beta}_0, \dots, \hat{\beta}_k$ are no longer optimal as some information ignored;
2. Statistical tests associated with the model (e.g., t-tests on the coefficients) are incorrect.
3. $p$-values for coefficients usually too small ("spurious regression").
4. AIC of fitted models misleading.

- Minimizing $\sum e_t^2$ avoids these problems.
- Maximizing likelihood is similar to minimizing $\sum e_t^2$.

# Estimation

If we minimize $\sum n_t^2$ (by using ordinary regression):

1. Estimated coefficients $\hat{\beta}_0, \dots, \hat{\beta}_k$ are no longer optimal as some information ignored;
2. Statistical tests associated with the model (e.g., t-tests on the coefficients) are incorrect.
3. *p*-values for coefficients usually too small ("spurious regression").
4. AIC of fitted models misleading.

- Minimizing $\sum e_t^2$ avoids these problems.
- Maximizing likelihood is similar to minimizing $\sum e_t^2$.

# Stationarity

## Model with ARIMA(1,1,1) errors

$$y_t = \beta_0 + \beta_1 x_{1,t} + \cdots + \beta_k x_{k,t} + n_t,$$
$$(1 - \phi_1 B)(1 - B)n_t = (1 + \theta_1 B)e_t,$$

## Equivalent to model with ARIMA(1,0,1) errors

$$y'_t = \beta_1 x'_{1,t} + \cdots + \beta_k x'_{k,t} + n'_t,$$
$$(1 - \phi_1 B)n'_t = (1 + \theta_1 B)e_t,$$

where $y'_t = y_t - y_{t-1}$, $x'_{t,i} = x_{t,i} - x_{t-1,i}$ and $n'_t = n_t - n_{t-1}$.

# Stationarity

## Model with ARIMA(1,1,1) errors

$$y_t = \beta_0 + \beta_1 x_{1,t} + \cdots + \beta_k x_{k,t} + n_t,$$
$$(1 - \phi_1 B)(1 - B)n_t = (1 + \theta_1 B)e_t,$$

## Equivalent to model with ARIMA(1,0,1) errors

$$y'_t = \beta_1 x'_{1,t} + \cdots + \beta_k x'_{k,t} + n'_t,$$
$$(1 - \phi_1 B)n'_t = (1 + \theta_1 B)e_t,$$

where $y'_t = y_t - y_{t-1}$, $x'_{t,i} = x_{t,i} - x_{t-1,i}$ and $n'_t = n_t - n_{t-1}$.

# Regression with ARIMA errors

Any regression with an ARIMA error can be rewritten as a regression with an ARMA error by differencing all variables with the same differencing operator as in the ARIMA model.

**Original data**

$$y_t = \beta_0 + \beta_1 x_{1,t} + \cdots + \beta_k x_{k,t} + n_t$$

$$\text{where} \quad \phi(B)(1-B)^d n_t = \theta(B) e_t$$

**After differencing all variables**

$$y'_t = \beta_1 x'_{1,t} + \cdots + \beta_k x'_{k,t} + n'_t.$$

$$\text{where} \quad \phi(B) n_t = \theta(B) e_t$$

$$\text{and} \quad y'_t = (1-B)^d y_t$$

# Regression with ARIMA errors

Any regression with an ARIMA error can be rewritten as a regression with an ARMA error by differencing all variables with the same differencing operator as in the ARIMA model.

## Original data

$$y_t = \beta_0 + \beta_1 x_{1,t} + \cdots + \beta_k x_{k,t} + n_t$$

$$\text{where} \quad \phi(B)(1 - B)^d n_t = \theta(B) e_t$$

## After differencing all variables

$$y'_t = \beta_1 x'_{1,t} + \cdots + \beta_k x'_{k,t} + n'_t.$$

$$\text{where} \quad \phi(B) n_t = \theta(B) e_t$$

$$\text{and} \quad y'_t = (1 - B)^d y_t$$

# Regression with ARIMA errors

Any regression with an ARIMA error can be rewritten as a regression with an ARMA error by differencing all variables with the same differencing operator as in the ARIMA model.

## Original data

$$y_t = \beta_0 + \beta_1 x_{1,t} + \cdots + \beta_k x_{k,t} + n_t$$

$$\text{where} \quad \phi(B)(1-B)^d n_t = \theta(B) e_t$$

## After differencing all variables

$$y'_t = \beta_1 x'_{1,t} + \cdots + \beta_k x'_{k,t} + n'_t.$$

$$\text{where} \quad \phi(B) n_t = \theta(B) e_t$$

$$\text{and} \quad y'_t = (1-B)^d y_t$$

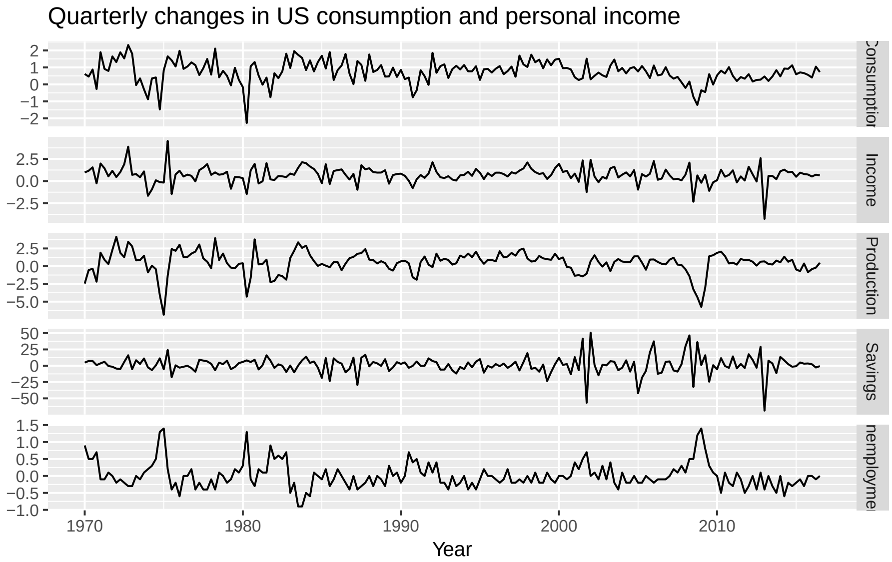
# Model selection

- Fit regression model with automatically selected ARIMA errors.
- Check that $e_t$ series looks like white noise.
- Note that estimation is done on the differenced series to ensure consistent estimators.

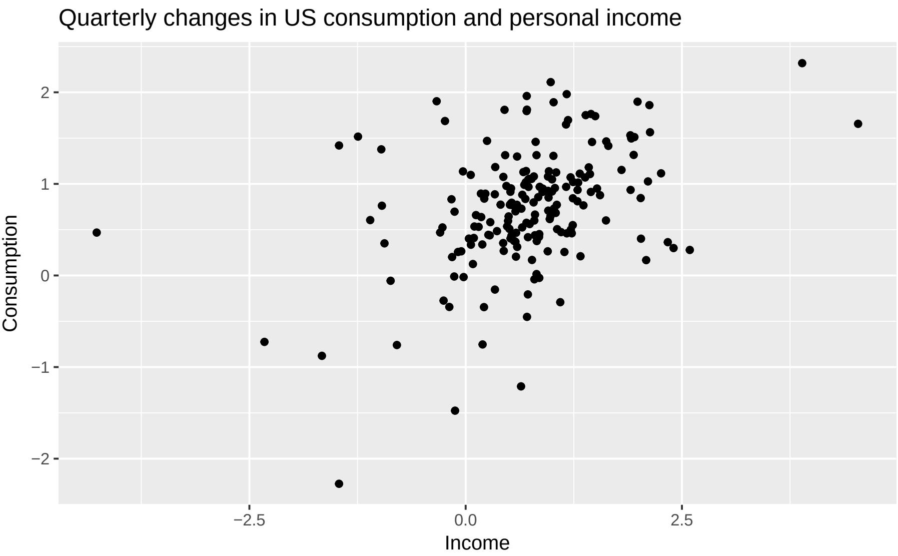
## Selecting predictors

- AICc can be calculated for final model.
- Repeat procedure for all subsets of predictors to be considered, and select model with lowest AICc value.

# US personal consumption & income



# US personal consumption & income



# US Personal Consumption and income

- No need for transformations or further differencing.
- Increase in income does not necessarily translate into instant increase in consumption (e.g., after the loss of a job, it may take a few months for expenses to be reduced to allow for the new circumstances). We will ignore this for now.

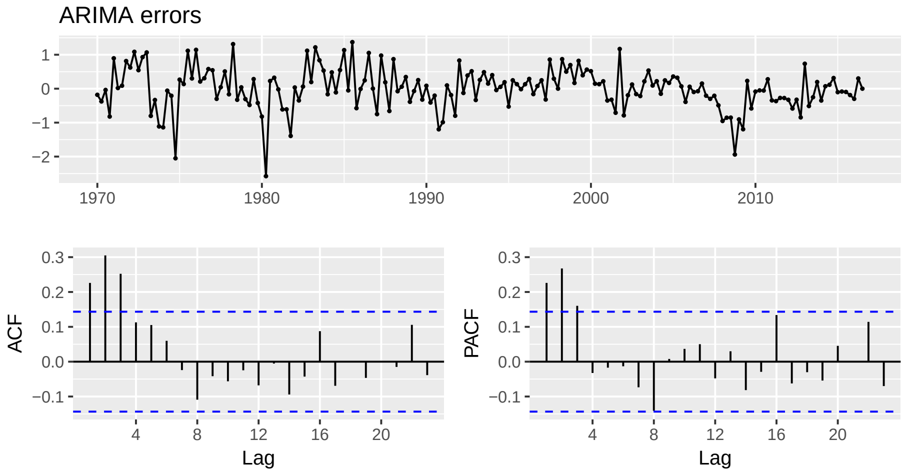
# US personal consumption & income

```r
(fit <- auto.arima(uschange[,"Consumption"],
    xreg=uschange[,"Income"]))
```

```
## Series: uschange[, "Consumption"]
## Regression with ARIMA(1,0,2) errors
##
## Coefficients:
##          ar1      ma1     ma2  intercept    xreg
##       0.6922  -0.5758  0.1984     0.5990  0.2028
## s.e.  0.1159   0.1301  0.0756     0.0884  0.0461
##
## sigma^2 estimated as 0.3219:  log likelihood=-156.95
## AIC=325.91   AICc=326.37   BIC=345.29
```

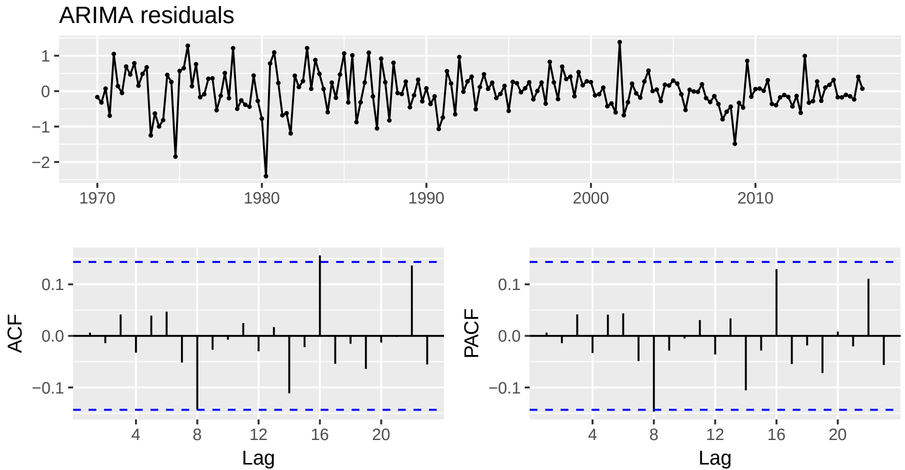
# US personal consumption & income

```
ggtsdisplay(residuals(fit, type='regression'),
  main="ARIMA errors")
```



# US personal consumption & income

```
ggtsdisplay(residuals(fit),
  main="ARIMA residuals")
```



# US Personal Consumption and Income

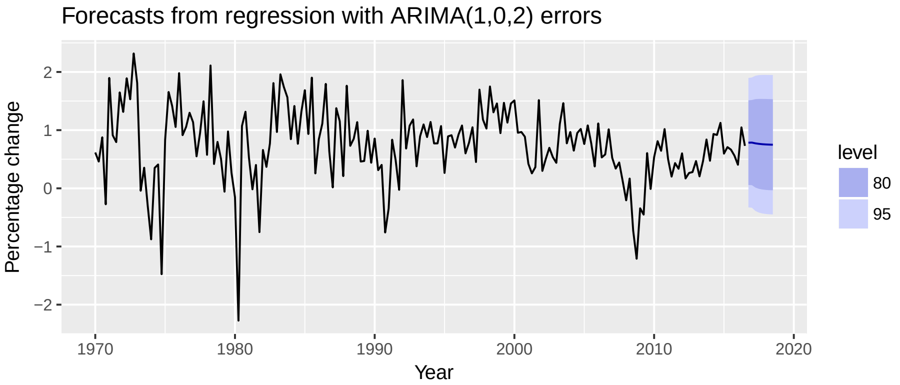
A Ljung-Box test shows the residuals are uncorrelated.

```
checkresiduals(fit, plot=FALSE)
```

```
##
##  Ljung-Box test
##
## data:  Residuals from Regression with ARIMA(1,0,2) err
## Q* = 5.8916, df = 3, p-value = 0.117
##
## Model df: 5.   Total lags used: 8
```

# US Personal Consumption and Income

```
fcast <- forecast(fit,
  xreg=rep(mean(uschange[,"Income"]),8), h=8)
autoplot(fcast) + xlab("Year") +
  ylab("Percentage change") +
  ggtitle("Forecasts from regression with ARIMA(1,0,2) errors"
```



# Forecasting

- To forecast a regression model with ARIMA errors, we need to forecast the regression part of the model and the ARIMA part of the model and combine the results.
  - Some explanatory variable are known into the future (e.g., time, dummies).
  - Separate forecasting models may be needed for other explanatory variables.

# Outline

# Stochastic & deterministic trends

## Deterministic trend

$$y_t = \beta_0 + \beta_1 t + n_t$$

where $n_t$ is ARMA process.

## Stochastic trend

$$y_t = \beta_0 + \beta_1 t + n_t$$

where $n_t$ is ARIMA process with $d \geq 1$.

Difference both sides until $n_t$ is stationary:

$$y'_t = \beta_1 + n'_t$$

where $n'_t$ is ARMA process.

# Stochastic & deterministic trends

## Deterministic trend

$$y_t = \beta_0 + \beta_1 t + n_t$$

where $n_t$ is ARMA process.

## Stochastic trend

$$y_t = \beta_0 + \beta_1 t + n_t$$

where $n_t$ is ARIMA process with $d \geq 1$.

Difference both sides until $n_t$ is stationary:

$$y'_t = \beta_1 + n'_t$$

where $n'_t$ is ARMA process.

# Stochastic & deterministic trends

## Deterministic trend

$$y_t = \beta_0 + \beta_1 t + n_t$$

where $n_t$ is ARMA process.

## Stochastic trend

$$y_t = \beta_0 + \beta_1 t + n_t$$

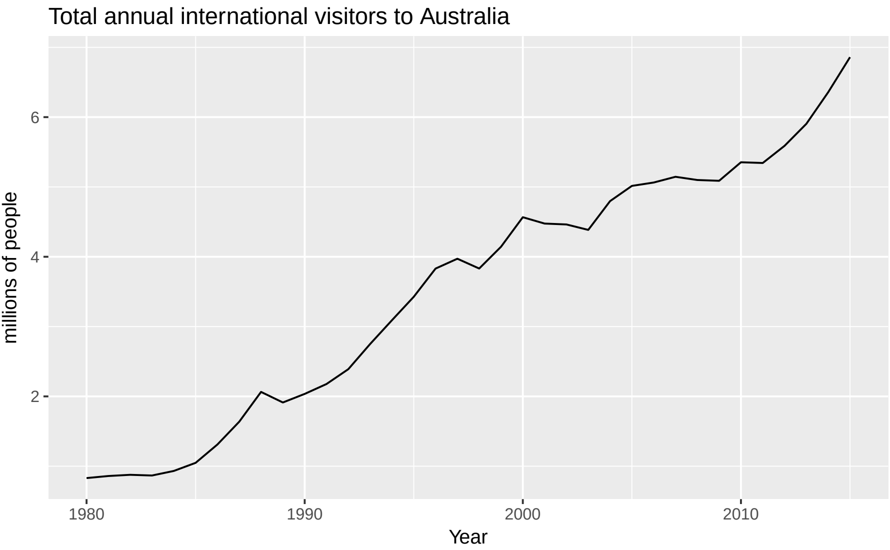
where $n_t$ is ARIMA process with $d \geq 1$.

Difference both sides until $n_t$ is stationary:

$$y'_t = \beta_1 + n'_t$$

where $n'_t$ is ARMA process.

# International visitors



# International visitors

## Deterministic trend

```
(fit1 <- auto.arima(austa, d=0, xreg=1:length(austa)))
```

```
## Series: austa
## Regression with ARIMA(2,0,0) errors
##
## Coefficients:
##          ar1      ar2  intercept    xreg
##       1.1127  -0.3805     0.4156  0.1710
## s.e.  0.1600   0.1585     0.1897  0.0088
##
## sigma^2 estimated as 0.02979:  log likelihood=13.6
## AIC=-17.2   AICc=-15.2   BIC=-9.28
```

$$y_t = 0.4173 + 0.1715t + n_t$$
$$n_t = 1.0371n_{t-1} - 0.3379n_{t-2} + e_t$$
$$e_t \sim \text{NID}(0, 0.02854).$$

# International visitors

## Deterministic trend

```r
(fit1 <- auto.arima(austa, d=0, xreg=1:length(austa)))
```

```
## Series: austa
## Regression with ARIMA(2,0,0) errors
##
## Coefficients:
##          ar1      ar2  intercept    xreg
##       1.1127  -0.3805     0.4156  0.1710
## s.e.  0.1600   0.1585     0.1897  0.0088
##
## sigma^2 estimated as 0.02979:  log likelihood=13.6
## AIC=-17.2   AICc=-15.2   BIC=-9.28
```

$$
\begin{aligned}
y_t &= 0.4173 + 0.1715t + n_t \\
n_t &= 1.0371n_{t-1} - 0.3379n_{t-2} + e_t \\
e_t &\sim \text{NID}(0, 0.02854).
\end{aligned}
$$

# International visitors

## Stochastic trend

```r
(fit2 <- auto.arima(austa,d=1))
```

```
## Series: austa
## ARIMA(0,1,1) with drift
##
## Coefficients:
##          ma1   drift
##       0.3006  0.1735
## s.e.  0.1647  0.0390
##
## sigma^2 estimated as 0.03376:  log likelihood=10.62
## AIC=-15.24   AICc=-14.46   BIC=-10.57
```

$$y_t - y_{t-1} = 0.1537 + e_t$$
$$y_t = y_0 + 0.1537t + n_t$$
$$n_t = n_{t-1} + e_t$$

# International visitors

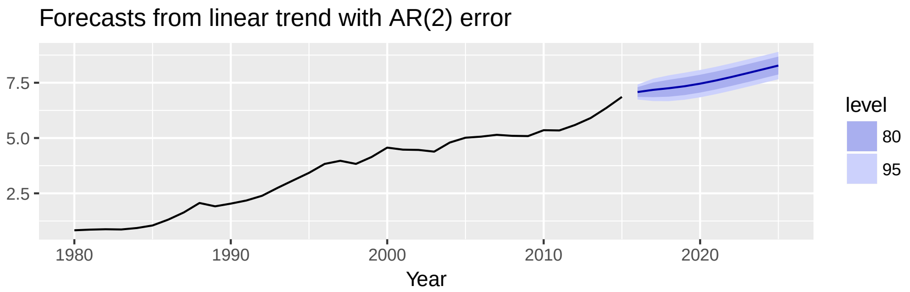
## Stochastic trend

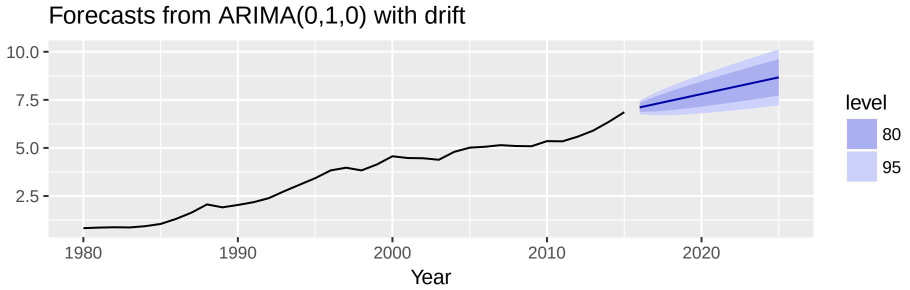
```
(fit2 <- auto.arima(austa,d=1))
```

```
## Series: austa
## ARIMA(0,1,1) with drift
##
## Coefficients:
##          ma1   drift
##       0.3006  0.1735
## s.e.  0.1647  0.0390
##
## sigma^2 estimated as 0.03376:  log likelihood=10.62
## AIC=-15.24   AICc=-14.46   BIC=-10.57
```

$$
\begin{aligned}
y_t - y_{t-1} &= 0.1537 + e_t \\
y_t &= y_0 + 0.1537t + n_t \\
n_t &= n_{t-1} + e_t
\end{aligned}
$$

# International visitors



# Forecasting with trend

- Point forecasts are almost identical, but prediction intervals differ.
- Stochastic trends have much wider prediction intervals because the errors are non-stationary.
- Be careful of forecasting with deterministic trends too far ahead.

# Outline

# Fourier terms for seasonality

Periodic seasonality can be handled using pairs of Fourier terms:

$$s_k(t) = \sin\left(\frac{2\pi kt}{m}\right) \qquad c_k(t) = \cos\left(\frac{2\pi kt}{m}\right)$$

$$y_t = \sum_{k=1}^{K} [\alpha_k s_k(t) + \beta_k c_k(t)] + n_t$$

- $n_t$ is non-seasonal ARIMA process.
- Every periodic function can be approximated by sums of sin and cos terms for large enough $K$.
- Choose $K$ by minimizing AICc.

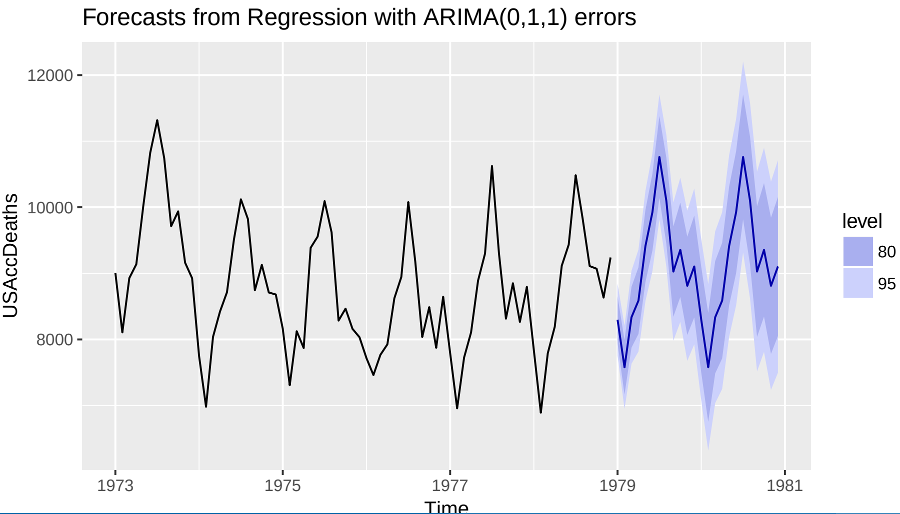
# US Accidental Deaths

```r
fit <- auto.arima(USAccDeaths,
         xreg=fourier(USAccDeaths, 5),
         seasonal=FALSE)

fc <- forecast(fit,
         xreg=fourier(USAccDeaths, 5, 24))
```

# US Accidental Deaths

```
autoplot(fc)
```



# Outline

# Lab Session 14

# Outline

# Dynamic regression models

## Sometimes a change in $x_t$ does not affect $y_t$ instantaneously

- $y_t$ = sales, $x_t$ = advertising.
- $y_t$ = stream flow, $x_t$ = rainfall.
- $y_t$ = size of herd, $x_t$ = breeding stock.

- These are dynamic systems with input ($x_t$) and output ($y_t$).
- $x_t$ is often a leading indicator.
- There can be multiple predictors.

# Dynamic regression models

## Sometimes a change in $x_t$ does not affect $y_t$ instantaneously

- $y_t$ = sales, $x_t$ = advertising.
- $y_t$ = stream flow, $x_t$ = rainfall.
- $y_t$ = size of herd, $x_t$ = breeding stock.

- These are dynamic systems with input ($x_t$) and output ($y_t$).
- $x_t$ is often a leading indicator.
- There can be multiple predictors.

# Dynamic regression models

## Sometimes a change in $x_t$ does not affect $y_t$ instantaneously

- $y_t$ = sales, $x_t$ = advertising.
- $y_t$ = stream flow, $x_t$ = rainfall.
- $y_t$ = size of herd, $x_t$ = breeding stock.

- These are dynamic systems with input ($x_t$) and output ($y_t$).
- $x_t$ is often a leading indicator.
- There can be multiple predictors.

# Lagged explanatory variables

The model include present and past values of predictor: $x_t, x_{t-1}, x_{t-2}, \ldots.$

$$y_t = a + \nu_0 x_t + \nu_1 x_{t-1} + \cdots + \nu_k x_{t-k} + n_t$$

where $n_t$ is an ARIMA process.

**Rewrite model as**

$$\begin{aligned} y_t &= a + (\nu_0 + \nu_1 B + \nu_2 B^2 + \cdots + \nu_k B^k) x_t + n_t \\ &= a + \nu(B) x_t + n_t. \end{aligned}$$

- $\nu(B)$ is called a *transfer function* since it describes how change in $x_t$ is transferred to $y_t$.
- $x$ can influence $y$, but $y$ is not allowed to influence $x$.

# Lagged explanatory variables

The model include present and past values of predictor: $x_t, x_{t-1}, x_{t-2}, \ldots.$

$$y_t = a + \nu_0 x_t + \nu_1 x_{t-1} + \cdots + \nu_k x_{t-k} + n_t$$

where $n_t$ is an ARIMA process.

**Rewrite model as**

$$\begin{aligned} y_t &= a + (\nu_0 + \nu_1 B + \nu_2 B^2 + \cdots + \nu_k B^k) x_t + n_t \\ &= a + \nu(B) x_t + n_t. \end{aligned}$$

- $\nu(B)$ is called a *transfer function* since it describes how change in $x_t$ is transferred to $y_t$.
- $x$ can influence $y$, but $y$ is not allowed to influence $x$.

# Lagged explanatory variables

The model include present and past values of predictor: $x_t, x_{t-1}, x_{t-2}, \ldots.$

$$y_t = a + \nu_0 x_t + \nu_1 x_{t-1} + \cdots + \nu_k x_{t-k} + n_t$$

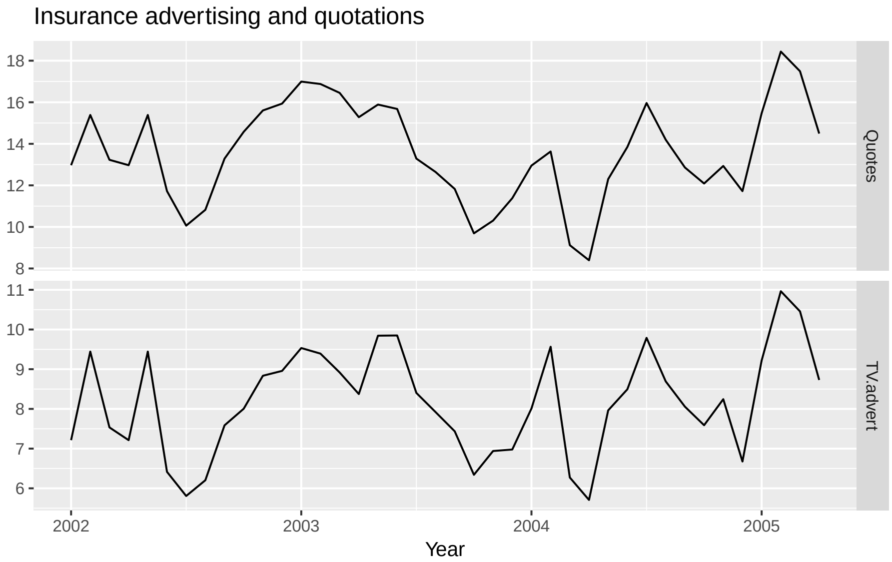
where $n_t$ is an ARIMA process.

## Rewrite model as

$$\begin{aligned} y_t &= a + (\nu_0 + \nu_1 B + \nu_2 B^2 + \cdots + \nu_k B^k) x_t + n_t \\ &= a + \nu(B) x_t + n_t. \end{aligned}$$

- $\nu(B)$ is called a *transfer function* since it describes how change in $x_t$ is transferred to $y_t$.
- $x$ can influence $y$, but $y$ is not allowed to influence $x$.

# Example: Insurance quotes and TV adverts



# Example: Insurance quotes and TV adverts

```r
Advert <- cbind(insurance[,2], c(NA,insurance[1:39,2]))
colnames(Advert) <- paste("AdLag",0:1,sep="")
(fit <- auto.arima(insurance[,1], xreg=Advert, d=0))
```

```
## Series: insurance[, 1]
## Regression with ARIMA(3,0,0) errors
##
## Coefficients:
##          ar1      ar2     ar3  intercept  AdLag0  AdLag1
##       1.4117  -0.9317  0.3591     2.0393  1.2564  0.1625
## s.e.  0.1698   0.2545  0.1592     0.9931  0.0667  0.0591
##
## sigma^2 estimated as 0.2165:  log likelihood=-23.89
## AIC=61.78   AICc=65.28   BIC=73.6
```

$$y_t = 2.04 + 1.26x_t + 0.16x_{t-1} + n_t$$
$$n_t = 1.41n_{t-1} - 0.93n_{t-2} + 0.36n_{t-3} + e_t$$

# Example: Insurance quotes and TV adverts

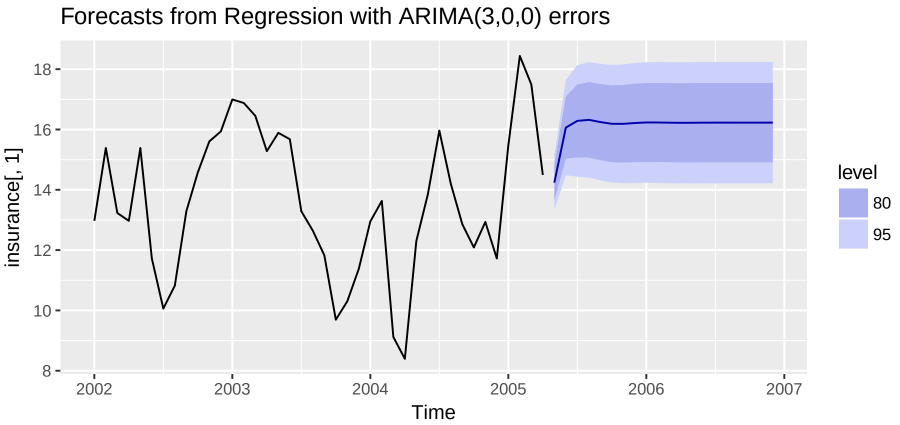
```r
Advert <- cbind(insurance[,2], c(NA,insurance[1:39,2]))
colnames(Advert) <- paste("AdLag",0:1,sep="")
(fit <- auto.arima(insurance[,1], xreg=Advert, d=0))
```

```
## Series: insurance[, 1]
## Regression with ARIMA(3,0,0) errors
##
## Coefficients:
##          ar1      ar2     ar3  intercept  AdLag0  AdLag1
##       1.4117  -0.9317  0.3591     2.0393  1.2564  0.1625
## s.e.  0.1698   0.2545  0.1592     0.9931  0.0667  0.0591
##
## sigma^2 estimated as 0.2165:  log likelihood=-23.89
## AIC=61.78   AICc=65.28   BIC=73.6
```

$$y_t = 2.04 + 1.26x_t + 0.16x_{t-1} + n_t$$
$$n_t = 1.41n_{t-1} - 0.93n_{t-2} + 0.36n_{t-3} + e_t$$

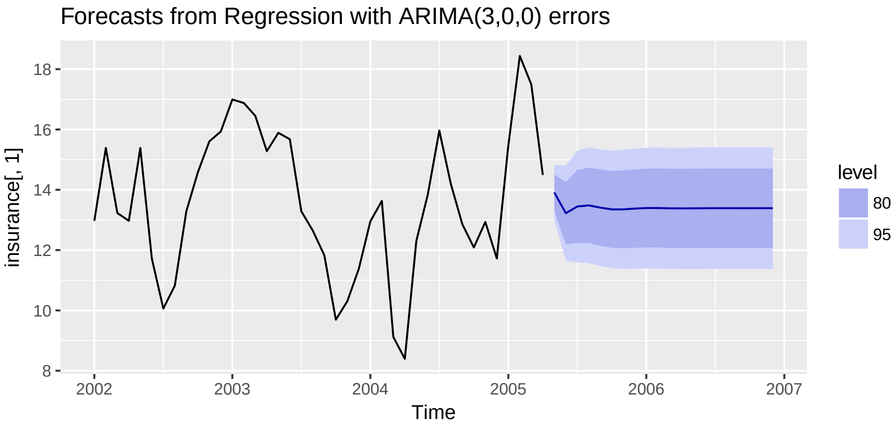
# Example: Insurance quotes and TV adverts

```
fc <- forecast(fit, h=20,
  xreg=cbind(c(Advert[40,1],rep(10,19)), rep(10,20)))
autoplot(fc)
```



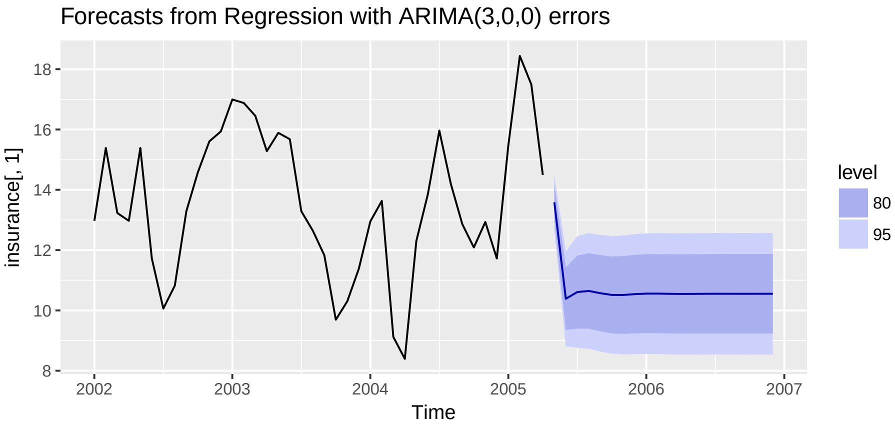
# Example: Insurance quotes and TV adverts

```r
fc <- forecast(fit, h=20,
  xreg=cbind(c(Advert[40,1],rep(8,19)), rep(8,20)))
autoplot(fc)
```



# Example: Insurance quotes and TV adverts

```
fc <- forecast(fit, h=20,
  xreg=cbind(c(Advert[40,1],rep(6,19)), rep(6,20)))
autoplot(fc)
```



# Dynamic regression models

$$y_t = a + \nu(B)x_t + n_t$$

where $n_t$ is an ARMA process. So

$$\phi(B)n_t = \theta(B)e_t \qquad \text{or} \qquad n_t = \frac{\theta(B)}{\phi(B)}e_t = \psi(B)e_t.$$

$$y_t = a + \nu(B)x_t + \psi(B)e_t$$

- ARMA models are rational approximations to general transfer functions of $e_t$.
- We can also replace $\nu(B)$ by a rational approximation.
- There is no R package for forecasting using a general transfer function approach.

# Dynamic regression models

$$y_t = a + \nu(B)x_t + n_t$$

where $n_t$ is an ARMA process. So

$$\phi(B)n_t = \theta(B)e_t \qquad \text{or} \qquad n_t = \frac{\theta(B)}{\phi(B)}e_t = \psi(B)e_t.$$

$$y_t = a + \nu(B)x_t + \psi(B)e_t$$

- ARMA models are rational approximations to general transfer functions of $e_t$.
- We can also replace $\nu(B)$ by a rational approximation.
- There is no R package for forecasting using a general transfer function approach.

# Dynamic regression models

$$y_t = a + \nu(B)x_t + n_t$$

where $n_t$ is an ARMA process. So

$$\phi(B)n_t = \theta(B)e_t \qquad \text{or} \qquad n_t = \frac{\theta(B)}{\phi(B)}e_t = \psi(B)e_t.$$

$$y_t = a + \nu(B)x_t + \psi(B)e_t$$

- ARMA models are rational approximations to general transfer functions of $e_t$.
- We can also replace $\nu(B)$ by a rational approximation.
- There is no R package for forecasting using a general transfer function approach.