3.1 Dynamic regression
Outline

1. Regression with ARIMA errors
2. Lab session 19
3. Some useful predictors for linear models
4. Dynamic harmonic regression
5. Lab session 20
6. Lagged predictors
Regression with ARIMA errors

Regression models

\[ y_t = \beta_0 + \beta_1 x_{1,t} + \cdots + \beta_k x_{k,t} + \varepsilon_t, \]

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

Example: ARIMA(1,1,1) errors

\[ y_t = \beta_0 + \beta_1 x_{1,t} + \cdots + \beta_k x_{k,t} + \eta_t, \]

\[ (1 - \phi_1 B)(1 - B) \eta_t = (1 + \theta_1 B) \varepsilon_t, \]

where \( \varepsilon_t \) is white noise.
Regression with ARIMA errors

Regression models

\[ y_t = \beta_0 + \beta_1 x_{1,t} + \cdots + \beta_k x_{k,t} + \varepsilon_t, \]

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

Example: ARIMA(1,1,1) errors

\[ y_t = \beta_0 + \beta_1 x_{1,t} + \cdots + \beta_k x_{k,t} + \eta_t, \]
\[ (1 - \phi_1 B)(1 - B)\eta_t = (1 + \theta_1 B)\varepsilon_t, \]

where \( \varepsilon_t \) is white noise.
Residuals and errors

Example: $\eta_t = \text{ARIMA}(1,1,1)$

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

$$(1 - \phi_1 B)(1 - B)\eta_t = (1 + \theta_1 B)\varepsilon_t,$$
Example: $\eta_t = \text{ARIMA}(1,1,1)$

\[
y_t = \beta_0 + \beta_1 x_{1,t} + \cdots + \beta_k x_{k,t} + \eta_t, \\
(1 - \phi_1 B)(1 - B)\eta_t = (1 + \theta_1 B)\varepsilon_t,
\]

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

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

1. Estimated coefficients $\hat{\beta}_0, \ldots, \hat{\beta}_k$ 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.
If we minimize $\sum \eta_t^2$ (by using ordinary regression):

1. Estimated coefficients $\hat{\beta}_0, \ldots, \hat{\beta}_k$ 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 $\sum \varepsilon_t^2$ avoids these problems.
Maximizing likelihood is similar to minimizing $\sum \varepsilon_t^2$. 
Regression with ARMA errors

\[ y_t = \beta_0 + \beta_1 x_{1,t} + \cdots + \beta_k x_{k,t} + \eta_t, \]

where \( \eta_t \) is an ARMA process.

- If we estimate the model while any variable is non-stationary, the estimated coefficients can be incorrect.
- Difference variables until all stationary.
- If necessary, apply same differencing to all variables to preserve interpretability.
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:

\[ y_t = \beta_0 + \beta_1 x_1, t + \ldots + \beta_k x_k, t + \eta_t \]

After differencing all variables:

\[ y'_t = \beta_1 x'_1, t + \ldots + \beta_k x'_k, t + \eta'_t \]

where \( \phi(B)(1-B)^d \eta_t = \theta(B) \epsilon_t \) and \( y'_t = (1-B)^d y_t \)
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

\[ y_t = \beta_0 + \beta_1 x_{1,t} + \cdots + \beta_k x_{k,t} + \eta_t \]

where \( \phi(B)(1 - B)^d \eta_t = \theta(B)\varepsilon_t \)
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**

\[ y_t = \beta_0 + \beta_1 x_{1,t} + \cdots + \beta_k x_{k,t} + \eta_t \]

where \( \phi(B)(1 - B)^d \eta_t = \theta(B) \varepsilon_t \)

<table>
<thead>
<tr>
<th>After differencing all variables</th>
</tr>
</thead>
<tbody>
<tr>
<td>[ y'<em>t = \beta_1 x'</em>{1,t} + \cdots + \beta_k x'_{k,t} + \eta'_t. ]</td>
</tr>
<tr>
<td>where ( \phi(B) \eta'_t = \theta(B) \varepsilon_t )</td>
</tr>
<tr>
<td>and ( y'_t = (1 - B)^d y_t )</td>
</tr>
</tbody>
</table>
Model selection

- Fit regression model with automatically selected ARIMA errors.
- Check that $\varepsilon_t$ series looks like white noise.

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 and income

```r
autoplot(uschange[,1:2], facets=TRUE) + xlab("Year") + ylab(""") + ggttitle("Quarterly changes in US consumption and personal income")
```

Quarterly changes in US consumption and personal income

![Graph showing quarterly changes in US consumption and personal income from 1970 to 2010.](image)
US personal consumption and income

```r
qplot(Income, Consumption, data=as.data.frame(uschange)) +
ggtitle("Quarterly changes in US consumption and personal 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.
(fit <- \texttt{auto.arima}(usk\texttt{u}change[,1], \texttt{xreg}=usk\texttt{u}change[,2]))

## Series: us\texttt{c}h\texttt{a}\texttt{n}\texttt{c}e[, 1]
## Regression with \texttt{ARIMA}(1,0,2) errors
##
## Coefficients:
##
## ar1   ma1  ma2 intercept xreg
## 0.692 -0.576 0.198 0.599 0.203
## s.e. 0.116 0.130 0.076 0.088 0.046
##
## \texttt{sigma}^2 estimated as 0.322: log likelihood=-156.9
## AIC=325.9 AICc=326.4 BIC=345.3
US personal consumption and income

\[
\text{fit} \leftarrow \text{auto.arima}(\text{uschange[,1]}, \text{xreg}=\text{uschange[,2]})
\]

## Series: uschange[, 1]
## Regression with ARIMA(1,0,2) errors

## Coefficients:

\[
\begin{array}{cccccc}
\text{ar1} & \text{ma1} & \text{ma2} & \text{intercept} & \text{xreg} \\
0.692 & -0.576 & 0.198 & 0.599 & 0.203 \\
\text{s.e.} & 0.116 & 0.130 & 0.076 & 0.088 & 0.046 \\
\end{array}
\]

## \sigma^2 estimated as 0.322: log likelihood=-156.9
## AIC=325.9  AICc=326.4  BIC=345.3

Write down the equations for the fitted model.
US personal consumption and income

`check_residuals(fit, test=FALSE)`

Residuals from Regression with ARIMA(1,0,2) errors

ACF

Count

Lag

residuals
US personal consumption and income

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

##
## Ljung-Box test
##
## data: Residuals from Regression with ARIMA(1,0,2) errors
## Q* = 5.9, df = 3, p-value = 0.1
##
## Model df: 5. Total lags used: 8
US personal consumption and income

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

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 predictors are known into the future (e.g., time, dummies).
- Separate forecasting models may be needed for other predictors.
- Forecast intervals ignore the uncertainty in forecasting the predictors.
Model daily electricity demand as a function of temperature using quadratic regression with ARMA errors.

```r
qplot(elecdaily[,"Temperature"], elecdaily[,"Demand"] ) +
xlab("Temperature") + ylab("Demand")
```
Daily electricity demand

```r
autoplot(elecdaily, facets = TRUE)
```
Daily electricity demand

```r
xreg <- cbind(MaxTemp = elecdaily[, "Temperature"],
               MaxTempSq = elecdaily[, "Temperature"]^2,
               Workday = elecdaily[, "WorkDay"])

fit <- auto.arima(elecdaily[, "Demand"], xreg = xreg)
checkresiduals(fit)
```

![Residuals from Regression with ARIMA(2,1,2)(2,0,0)[7] errors](image_url)
## Ljung-Box test

### data: Residuals from Regression with ARIMA(2,1,2)(2,0,0)[7] errors
### $Q^* = 28$, $df = 4$, p-value = $1e^{-05}$

### Model df: 10. Total lags used: 14
Daily electricity demand

# Forecast one day ahead

```r
forecast(fit, xreg = cbind(26, 26^2, 1))
```

```r
## Point Forecast Lo 80 Hi 80 Lo 95 Hi 95
## 53.14 189.8 181.3 198.2 176.8 202.7
```
Daily electricity demand

\[
\text{fcast } \leftarrow \text{forecast}(\text{fit}, \\
\text{xreg } = \text{cbind(rep(26,14), rep(26^2,14),} \\
\text{c(0,1,0,0,1,1,1,1,0,0,1,1,1)))} \\
\text{autoplot(fcast) + ylab("Electricity demand (GW)"))}
\]
Outline

1. Regression with ARIMA errors
2. Lab session 19
3. Some useful predictors for linear models
4. Dynamic harmonic regression
5. Lab session 20
6. Lagged predictors
Lab Session 19
1. Regression with ARIMA errors
2. Lab session 19
3. Some useful predictors for linear models
4. Dynamic harmonic regression
5. Lab session 20
6. Lagged predictors
Linear trend

\[ x_t = t \]

- \( t = 1, 2, \ldots, T \)
- Strong assumption that trend will continue.
If a categorical variable takes only two values (e.g., ‘Yes’ or ‘No’), then an equivalent numerical variable can be constructed taking value 1 if yes and 0 if no. This is called a dummy variable.
If there are more than two categories, then the variable can be coded using several dummy variables (one fewer than the total number of categories).
Beware of the dummy variable trap!

- Using one dummy for each category gives too many dummy variables!
- The regression will then be singular and inestimable.
- Either omit the constant, or omit the dummy for one category.
- The coefficients of the dummies are relative to the omitted category.
Uses of dummy variables

Seasonal dummies

- For quarterly data: use 3 dummies
- For monthly data: use 11 dummies
- For daily data: use 6 dummies
- What to do with weekly data?

Outliers

If there is an outlier, you can use a dummy variable (taking value 1 for that observation and 0 elsewhere) to remove its effect.

Public holidays

For daily data: if it is a public holiday, dummy=1; otherwise dummy=0.
Uses of dummy variables

Seasonal dummies

- For quarterly data: use 3 dummies
- For monthly data: use 11 dummies
- For daily data: use 6 dummies
- What to do with weekly data?

Outliers

- If there is an outlier, you can use a dummy variable (taking value 1 for that observation and 0 elsewhere) to remove its effect.
Uses of dummy variables

Seasonal dummies

- For quarterly data: use 3 dummies
- For monthly data: use 11 dummies
- For daily data: use 6 dummies
- What to do with weekly data?

Outliers

- If there is an outlier, you can use a dummy variable (taking value 1 for that observation and 0 elsewhere) to remove its effect.

Public holidays

- For daily data: if it is a public holiday, dummy=1, otherwise dummy=0.
Fourier series

Periodic seasonality can be handled using pairs of Fourier terms:

\[ s_k(t) = \sin \left( \frac{2\pi kt}{m} \right) \quad c_k(t) = \cos \left( \frac{2\pi kt}{m} \right) \]

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

- Every periodic function can be approximated by sums of sin and cos terms for large enough \( K \).
- Choose \( K \) by minimizing AICc.
- Called “harmonic regression”
- `fourier()` function generates these.
Intervention variables

Spikes

- Equivalent to a dummy variable for handling an outlier.
Intervention variables

Spikes

■ Equivalent to a dummy variable for handling an outlier.

Steps

■ Variable takes value 0 before the intervention and 1 afterwards.
Intervention variables

Spikes

- Equivalent to a dummy variable for handling an outlier.

Steps

- Variable takes value 0 before the intervention and 1 afterwards.

Change of slope

- Variables take values 0 before the intervention and values \( \{1, 2, 3, \ldots \} \) afterwards.
Holidays

For monthly data

- Christmas: always in December so part of monthly seasonal effect
- Easter: use a dummy variable $v_t = 1$ if any part of Easter is in that month, $v_t = 0$ otherwise.
- Ramadan and Chinese new year similar.
With monthly data, if the observations vary depending on how many different types of days in the month, then trading day predictors can be useful.

\[ z_1 = \# \text{ Mondays in month}; \]
\[ z_2 = \# \text{ Tuesdays in month}; \]
\[ \vdots \]
\[ z_7 = \# \text{ Sundays in month}. \]
Distributed lags

Lagged values of a predictor.

Example: $x$ is advertising which has a delayed effect

$$x_1 = \text{advertising for previous month};$$
$$x_2 = \text{advertising for two months previously};$$
$$\vdots$$
$$x_m = \text{advertising for } m \text{ months previously}.$$
Outline

1. Regression with ARIMA errors
2. Lab session 19
3. Some useful predictors for linear models
4. Dynamic harmonic regression
5. Lab session 20
6. Lagged predictors
Dynamic harmonic regression

Combine Fourier terms with ARIMA errors

**Advantages**
- it allows any length seasonality;
- for data with more than one seasonal period, you can include Fourier terms of different frequencies;
- the seasonal pattern is smooth for small values of $K$ (but more wiggly seasonality can be handled by increasing $K$);
- the short-term dynamics are easily handled with a simple ARMA error.

**Disadvantages**
- seasonality is assumed to be fixed
Eating-out expenditure

```r
cafe04 <- window(auscafe, start=2004)
autoplot(cafe04)
```
Eating-out expenditure

Regression with ARIMA(3, 1, 4) errors and $\lambda = 0$

K= 1    AICC= −560.97
Eating-out expenditure

Regression with ARIMA(3, 1, 2) errors and $\lambda = 0$

K = 2  AICC = -617.16
Eating-out expenditure

Regression with ARIMA(2, 1, 0) errors and $\lambda = 0$

$K = 3$  $AICC = -693.31$
Eating-out expenditure

Regression with ARIMA(5, 1, 0) errors and $\lambda = 0$

K = 4  AICC = −790.74
Regression with ARIMA(0, 1, 1) errors and $\lambda = 0$
Eating-out expenditure

Regression with ARIMA(0, 1, 1) errors and $\lambda = 0$

K = 6    AICc = −827.3
Eating-out expenditure

```r
fit <- auto.arima(cafe04, xreg=fourier(cafe04, K=5), seasonal = FALSE, lambda = 0)
fc <- forecast(fit, xreg=fourier(cafe04, K=5, h=24))
autoplot(fc)
```
Example: weekly gasoline products

```r
harmonics <- fourier(gasoline, K = 13)
(fit <- auto.arima(gasoline, xreg = harmonics, seasonal = FALSE))
```

```
## Series: gasoline
## Regression with ARIMA(0,1,2) errors
##
## Coefficients:
##     ma1   ma2  drift    S1-52     C1-52     S2-52
##         -0.961 0.094 0.001  0.031   -0.255  -0.052
##     s.e.  0.027 0.029 0.001  0.012    0.012   0.009
##     C2-52    S3-52    C3-52    S4-52    C4-52    S5-52
##         -0.017 0.024 -0.099  0.032   -0.026  -0.001
##     s.e.  0.009  0.008  0.008  0.008    0.008    0.008
##     C5-52    S6-52    C6-52    S7-52    C7-52    S8-52
##         -0.047 0.058 -0.032  0.028   -0.026  -0.001
##     s.e.  0.008  0.008  0.008  0.008    0.008    0.008
##     C8-52    S9-52    C9-52    S10-52   C10-52   S11-52
##          0.014 -0.017  0.012  -0.024    0.023    0.000
##     s.e.  0.008  0.008  0.008  0.008    0.008    0.008
##     C11-52   S12-52   C12-52   S13-52   C13-52
##         -0.019 -0.029 -0.018   0.001   -0.018
##     s.e.  0.008  0.008  0.008  0.008    0.008    0.008
##
## sigma^2 estimated as 0.056:  log likelihood=43.66
## AIC=-27.33  AICc=-25.92  BIC=129
```
Example: weekly gasoline products

```r
checkresiduals(fit, test=FALSE)
```

Residuals from Regression with ARIMA(0,1,2) errors

Lag

ACF

residuals

count
Example: weekly gasoline products

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

```
##
## Ljung-Box test
##
## data:  Residuals from Regression with ARIMA(0,1,2) errors
## Q* = 130, df = 75, p-value = 6e-05
##
## Model df: 29.  Total lags used: 104.357142857143
```
Example: weekly gasoline products

newharmonics <- `fourier`(gasoline, K = 13, h = 156)
fc <- `forecast`(fit, xreg = newharmonics)
`autoplot`(fc)
5-minute call centre volume

**autoplot**(calls)
xreg <- `fourier`(calls, \[K = c(10,0)\])

(\text{fit} <- \text{auto.arima}(\text{calls, xreg=xreg, seasonal=FALSE, stationary=TRUE}))

## Series: calls
## Regression with ARIMA(3,0,2) errors
##
## Coefficients:
##            ar1    ar2   ar3    ma1    ma2 intercept
## s.e. 0.169 0.178 0.013 0.169 0.137      1.764
## S1-169 C1-169 S2-169 C2-169 S3-169
## s.e. 0.701 0.701 0.379 0.379 0.273
## C3-169 S4-169 C4-169 S5-169 C5-169 S6-169
## -9.327 -9.532 -2.797 -2.239  2.893    0.173
## s.e. 0.273 0.223 0.223 0.196  0.196    0.179
## C6-169 S7-169 C7-169 S8-169 C8-169 S9-169
##  3.305  0.855  0.294  0.857 -1.391   -0.986
## s.e. 0.179 0.168 0.168 0.160  0.160    0.155
## C9-169 S10-169 C10-169
## -0.345 -1.196  0.801
## s.e. 0.155 0.150 0.150
##
## \sigma^2 estimated as 243:  log likelihood=-115412
## AIC=230877  AICc=230877  BIC=231099
5-minute call centre volume

`checkresiduals(fit, test=FALSE)`

Residuals from Regression with ARIMA(3,0,2) errors

ACF

Lag

Count

residuals

52
fc <- `forecast`(fit, `xreg = fourier`(calls, c(10,0), 1690))
`autoplot`(fc)
Outline

1. Regression with ARIMA errors
2. Lab session 19
3. Some useful predictors for linear models
4. Dynamic harmonic regression
5. Lab session 20
6. Lagged predictors
Lab Session 20
Outline

1. Regression with ARIMA errors
2. Lab session 19
3. Some useful predictors for linear models
4. Dynamic harmonic regression
5. Lab session 20
6. Lagged predictors
Lagged predictors

Sometimes a change in $x_t$ does not affect $y_t$ instantaneously
Lagged predictors

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

- $y_t = \text{sales, } x_t = \text{advertising.}$
- $y_t = \text{stream flow, } x_t = \text{rainfall.}$
- $y_t = \text{size of herd, } x_t = \text{breeding stock.}$
Lagged predictors

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

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

- 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 predictors

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

\[ y_t = \alpha + \nu_0 x_t + \nu_1 x_{t-1} + \cdots + \nu_k x_{t-k} + \eta_t \]

where $\eta_t$ is an ARIMA process.
Lagged predictors

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

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

where $\eta_t$ is an ARIMA process.

Rewrite model as

$$y_t = a + (\nu_0 + \nu_1 B + \nu_2 B^2 + \ldots + \nu_k B^k)x_t + \eta_t$$

$$= a + \nu (B)x_t + \eta_t.$$
Lagged predictors

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

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

where $\eta_t$ is an ARIMA process.

Rewrite model as

$$y_t = a + (\nu_0 + \nu_1 B + \nu_2 B^2 + \cdots + \nu_k B^k) x_t + \eta_t$$

$$= a + \nu(B) x_t + \eta_t.$$  

$\nu(B)$ 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 $x$. 

58
Example: Insurance quotes and TV adverts

```r
autoplot(insurance, facets=TRUE) +
  xlab("Year") + ylab("") +
  ggtitle("Insurance advertising and quotations")
```

Insurance advertising and quotations

Quotes

TV.advert

Year

2002
2003
2004
2005
Example: Insurance quotes and TV adverts

Advert <- cbind(
  AdLag0 = insurance[, "TV.advert"],
  AdLag1 = lag(insurance[, "TV.advert"], -1),
  AdLag2 = lag(insurance[, "TV.advert"], -2),
  AdLag3 = lag(insurance[, "TV.advert"], -3)) %>%
  head(NROW(insurance))

# Restrict data so models use same fitting period
fit1 <- auto.arima(insurance[4:40, 1], xreg=Advert[4:40, 1], stationary=TRUE)
fit2 <- auto.arima(insurance[4:40, 1], xreg=Advert[4:40, 1:2], stationary=TRUE)
fit3 <- auto.arima(insurance[4:40, 1], xreg=Advert[4:40, 1:3], stationary=TRUE)
fit4 <- auto.arima(insurance[4:40, 1], xreg=Advert[4:40, 1:4], stationary=TRUE)
c(fit1$aicc, fit2$aicc, fit3$aicc, fit4$aicc)

## [1] 68.50 60.02 62.83 68.02
Example: Insurance quotes and TV adverts

```r
(fit <- auto.arima(insurance[,1], xreg=Advert[,1:2], stationary=TRUE))
```

```r
## Series: insurance[, 1]
## Regression with ARIMA(3,0,0) errors
##
## Coefficients:
##   ar1  ar2  ar3 intercept AdLag0  AdLag1
## 1.412 -0.932 0.359    2.039   1.256   0.162
## s.e. 0.170 0.255 0.159  0.993   0.067   0.059
##
## sigma^2 estimated as 0.217:  log likelihood=-23.89
## AIC=61.78   AICc=65.28   BIC=73.6
```
Example: Insurance quotes and TV adverts

```r
(fit <- auto.arima(insurance[,1], xreg=Advert[,1:2], stationary=TRUE))
```

```r
## Series: insurance[, 1]
## Regression with ARIMA(3,0,0) errors
##
## Coefficients:
## ar1  ar2  ar3 intercept AdLag0  AdLag1
## 1.412 -0.932  0.359   2.039   1.256  0.162
## s.e. 0.170 0.255 0.159  0.993  0.067  0.059
##
## sigma^2 estimated as 0.217: log likelihood=-23.89
## AIC=61.78  AICc=65.28  BIC=73.6
```

\[ y_t = 2.04 + 1.26x_t + 0.16x_{t-1} + \eta_t, \]

\[ \eta_t = 1.41\eta_{t-1} - 0.93\eta_{t-2} + 0.36\eta_{t-3} + \varepsilon_t, \]
Example: Insurance quotes and TV adverts

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

Forecasts from Regression with ARIMA(3,0,0) errors
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)
```

Forecasts from Regression with ARIMA(3,0,0) errors
Example: Insurance quotes and TV adverts

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

![Forecasts from Regression with ARIMA(3,0,0) errors](image)