



# ETC3550: Applied forecasting for business and economics

Ch5. Regression models

[OTexts.org/fpp2/](https://OTexts.org/fpp2/)

# Outline

- 1 The linear model with time series
- 2 Residual diagnostics
- 3 Some useful predictors for linear models
- 4 Selecting predictors and forecast evaluation
- 5 Forecasting with regression
- 6 Matrix formulation
- 7 Correlation, causation and forecasting

# Multiple regression and forecasting

$$y_t = \beta_0 + \beta_1 x_{1,t} + \beta_2 x_{2,t} + \cdots + \beta_k x_{k,t} + \varepsilon_t.$$

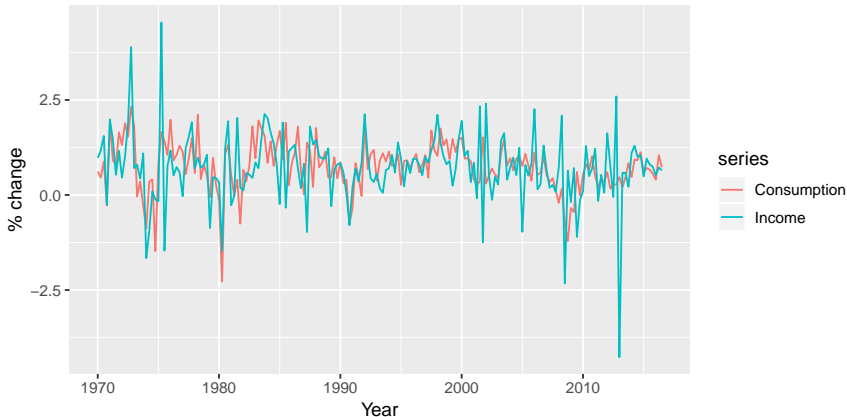
- $y_t$  is the variable we want to predict: the “response” variable
- Each  $x_{j,t}$  is numerical and is called a “predictor”. They are usually assumed to be known for all past and future times.
- The coefficients  $\beta_1, \dots, \beta_k$  measure the effect of each predictor after taking account of the effect of all other predictors in the model.

That is, the coefficients measure the **marginal effects**.

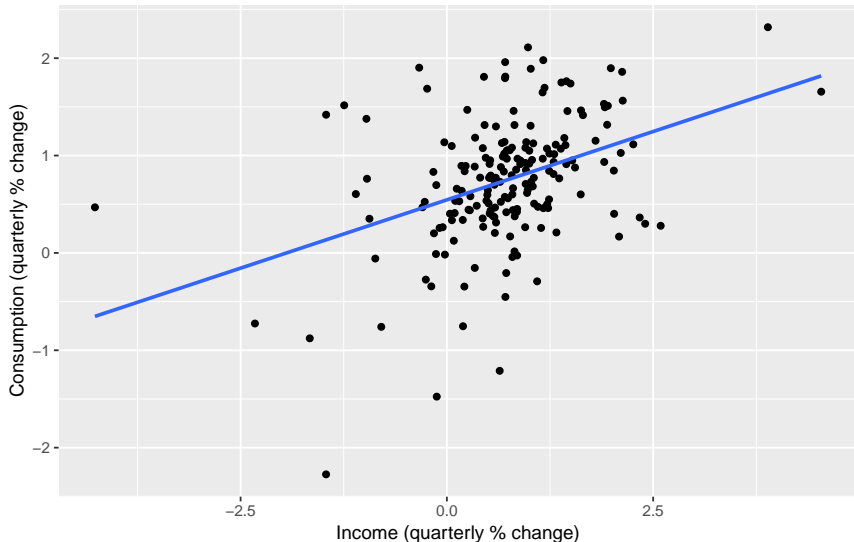
- $\varepsilon_t$  is a white noise error term

# Example: US consumption expenditure

```
autoplot(uschange[,c("Consumption","Income")]) +  
  ylab("% change") + xlab("Year")
```



# Example: US consumption expenditure

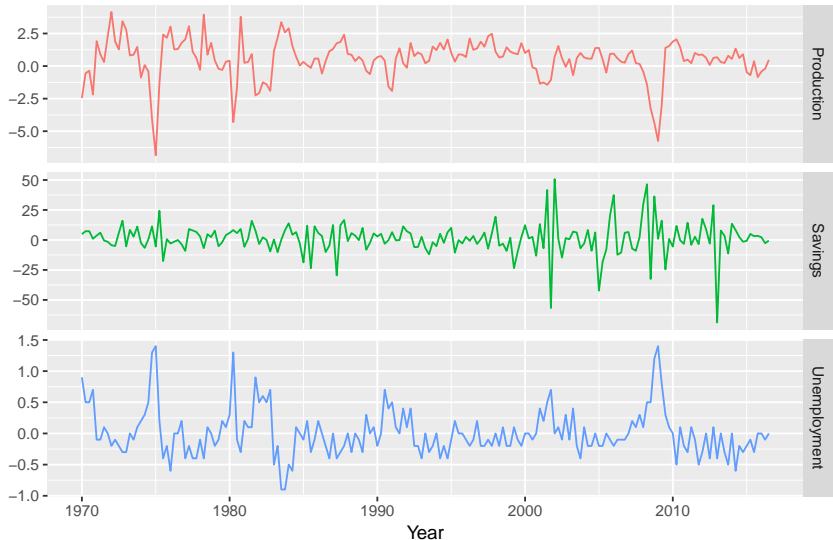


# Example: US consumption expenditure

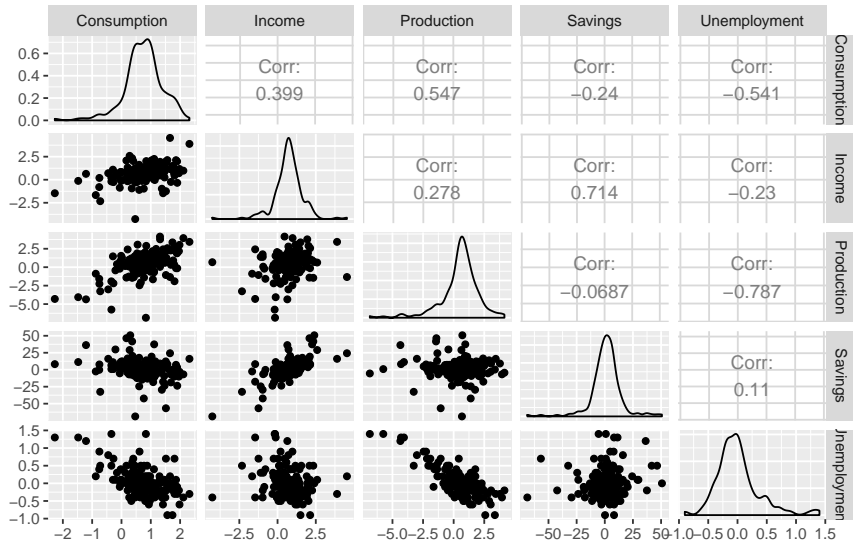
```
tslm(Consumption ~ Income, data=uschange) %>% summary
```

```
##
## Call:
## tslm(formula = Consumption ~ Income, data = uschange)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.40845 -0.31816  0.02558  0.29978  1.45157
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.54510     0.05569   9.789  < 2e-16 ***
## Income       0.28060     0.04744   5.915 1.58e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.6026 on 185 degrees of freedom
## Multiple R-squared:  0.159, Adjusted R-squared:  0.1545
## F-statistic: 34.98 on 1 and 185 DF, p-value: 1.577e-08
```

# Example: US consumption expenditure



# Example: US consumption expenditure



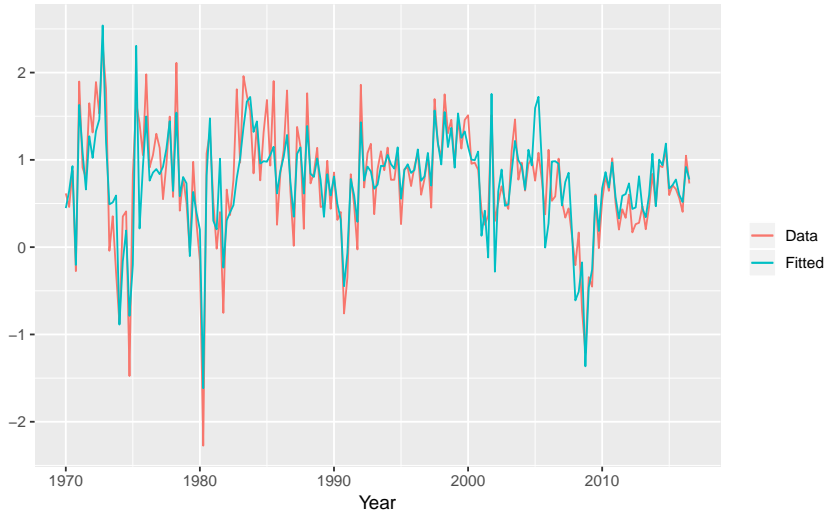


# Example: US consumption expenditure

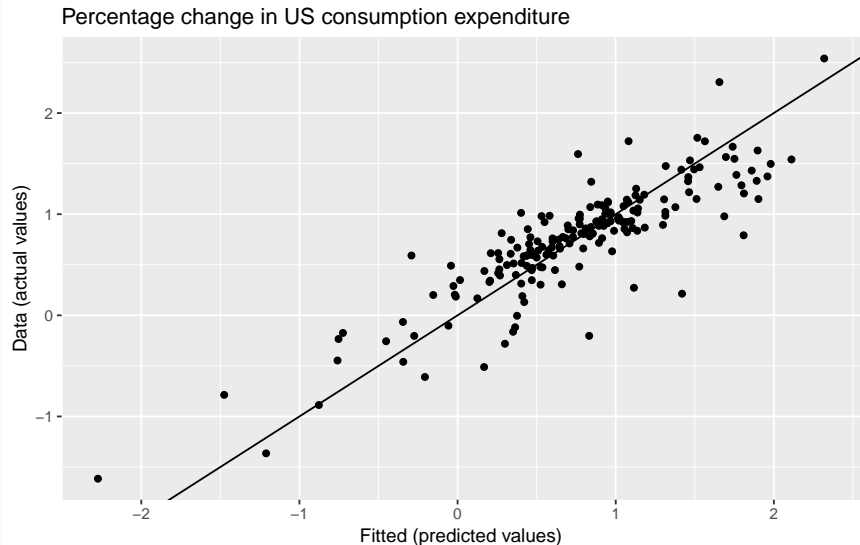
```
fit.consMR <- tslm(  
  Consumption ~ Income + Production + Unemployment + Savings,  
  data=uschange)  
summary(fit.consMR)  
  
##  
## Call:  
## tslm(formula = Consumption ~ Income + Production + Unemployment +  
##       Savings, data = uschange)  
##  
## Residuals:  
##      Min      1Q   Median      3Q      Max   
## -0.88296 -0.17638 -0.03679  0.15251  1.20553  
##  
## Coefficients:  
##              Estimate Std. Error t value Pr(>|t|)      
## (Intercept)   0.26729    0.03721   7.184 1.68e-11 ***  
## Income        0.71449    0.04219  16.934 < 2e-16 ***  
## Production    0.04589    0.02588   1.773  0.0778 .  
## Unemployment -0.20477    0.10550  -1.941  0.0538 .  
## Savings       -0.04527    0.00278 -16.287 < 2e-16 ***  
## ---  
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1  
##  
## Residual standard error: 0.3286 on 182 degrees of freedom  
## Multiple R-squared:  0.754, Adjusted R-squared:  0.7486  
## F-statistic: 139.5 on 4 and 182 DF,  p-value: < 2.2e-16
```

# Example: US consumption expenditure

Percentage change in US consumption expenditure

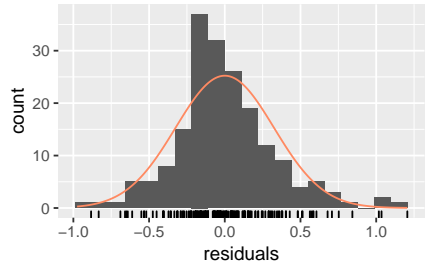
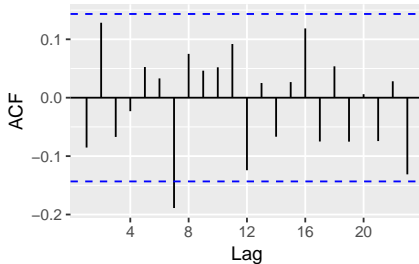
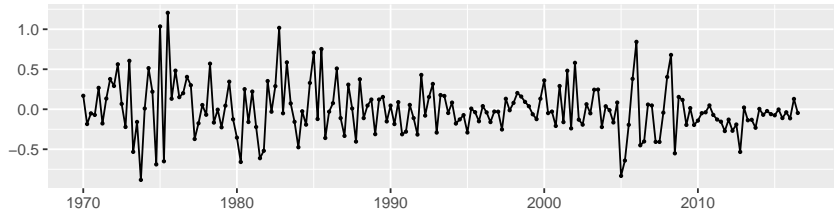


# Example: US consumption expenditure



# Example: US consumption expenditure

Residuals from Linear regression model



# Outline

- 1 The linear model with time series
- 2 Residual diagnostics
- 3 Some useful predictors for linear models
- 4 Selecting predictors and forecast evaluation
- 5 Forecasting with regression
- 6 Matrix formulation
- 7 Correlation, causation and forecasting

# Multiple regression and forecasting

For forecasting purposes, we require the following assumptions:

- $\varepsilon_t$  are uncorrelated and zero mean
- $\varepsilon_t$  are uncorrelated with each  $x_{j,t}$ .

# Multiple regression and forecasting

For forecasting purposes, we require the following assumptions:

- $\varepsilon_t$  are uncorrelated and zero mean
- $\varepsilon_t$  are uncorrelated with each  $x_{j,t}$ .

It is **useful** to also have  $\varepsilon_t \sim N(0, \sigma^2)$  when producing prediction intervals or doing statistical tests.

# Residual plots

Useful for spotting outliers and whether the linear model was appropriate.

- Scatterplot of residuals  $\varepsilon_t$  against each predictor  $x_{j,t}$ .
- Scatterplot residuals against the fitted values  $\hat{y}_t$
- Expect to see scatterplots resembling a horizontal band with no values too far from the band and no patterns such as curvature or increasing spread.



# Residual patterns

- If a plot of the residuals vs any predictor in the model shows a pattern, then the relationship is nonlinear.
- If a plot of the residuals vs any predictor **not** in the model shows a pattern, then the predictor should be added to the model.
- If a plot of the residuals vs fitted values shows a pattern, then there is heteroscedasticity in the errors. (Could try a transformation.)

# Breusch-Godfrey test

**OLS regression:**

$$y_t = \beta_0 + \beta_1 x_{t,1} + \dots + \beta_k x_{t,k} + u_t$$

**Auxiliary regression:**

$$\hat{u}_t = \beta_0 + \beta_1 x_{t,1} + \dots + \beta_k x_{t,k} + \rho_1 \hat{u}_{t-1} + \dots + \rho_p \hat{u}_{t-p} + \varepsilon_t$$

If  $R^2$  statistic is calculated for this model, then

$$(T - p)R^2 \sim \chi_p^2,$$

when there is no serial correlation up to lag  $p$ , and  $T$  = length of series.

- Breusch-Godfrey test better than Ljung-Box for regression models.

# US consumption again

```
##  
## Breusch-Godfrey test for serial correlation of order up to 8  
##  
## data: Residuals from Linear regression model  
## LM test = 14.874, df = 8, p-value = 0.06163
```

## If the model fails the Breusch-Godfrey test ...

- The forecasts are not wrong, but have higher variance than they need to.
- There is information in the residuals that we should exploit.
- This is done with a regression model with ARMA errors.

# Outline

- 1 The linear model with time series
- 2 Residual diagnostics
- 3 Some useful predictors for linear models**
- 4 Selecting predictors and forecast evaluation
- 5 Forecasting with regression
- 6 Matrix formulation
- 7 Correlation, causation and forecasting

## Linear trend

$$x_t = t$$

- $t = 1, 2, \dots, T$
- Strong assumption that trend will continue.

# Dummy variables

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**.

	A	B
1	Yes	1
2	Yes	1
3	No	0
4	Yes	1
5	No	0
6	No	0
7	Yes	1
8	Yes	1
9	No	0
10	No	0
11	No	0
12	No	0
13	Yes	1
14	No	0

# Dummy variables

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).

	A	B	C	D	E
1	Monday	1	0	0	0
2	Tuesday	0	1	0	0
3	Wednesday	0	0	1	0
4	Thursday	0	0	0	1
5	Friday	0	0	0	0
6	Monday	1	0	0	0
7	Tuesday	0	1	0	0
8	Wednesday	0	0	1	0
9	Thursday	0	0	0	1
10	Friday	0	0	0	0
11	Monday	1	0	0	0
12	Tuesday	0	1	0	0
13	Wednesday	0	0	1	0
14	Thursday	0	0	0	1
15	Friday	0	0	0	0

# 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?

# 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

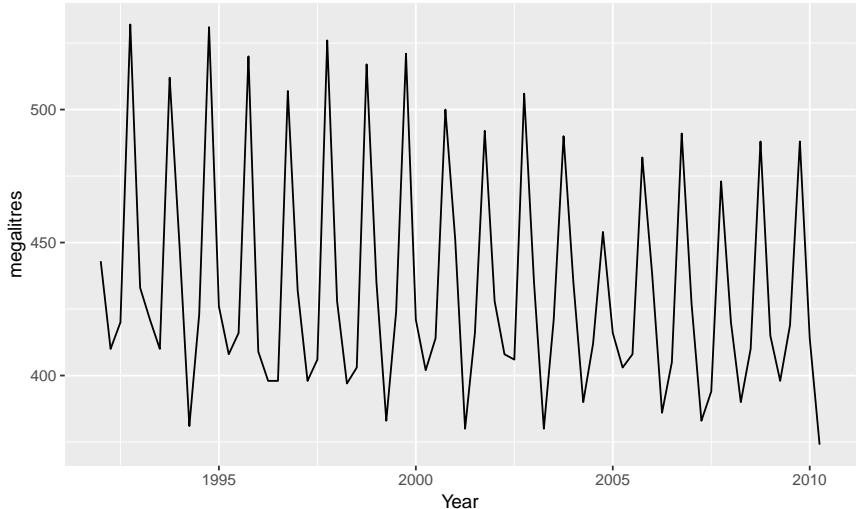
- If there is an outlier, you can use a dummy variable (taking value 1 for that observation and 0 elsewhere) to remove its effect.

## Public holidays

- For daily data: if it is a public holiday,  $\text{dummy}=1$ , otherwise  $\text{dummy}=0$ .

# Beer production revisited

Australian quarterly beer production



# Beer production revisited

## Regression model

$$y_t = \beta_0 + \beta_1 t + \beta_2 d_{2,t} + \beta_3 d_{3,t} + \beta_4 d_{4,t} + \varepsilon_t$$

- $d_{i,t} = 1$  if  $t$  is quarter  $i$  and 0 otherwise.

# Beer production revisited

```
fit.beer <- tslm(beer ~ trend + season)
summary(fit.beer)
```

```
##
## Call:
## tslm(formula = beer ~ trend + season)
##
## Residuals:
```

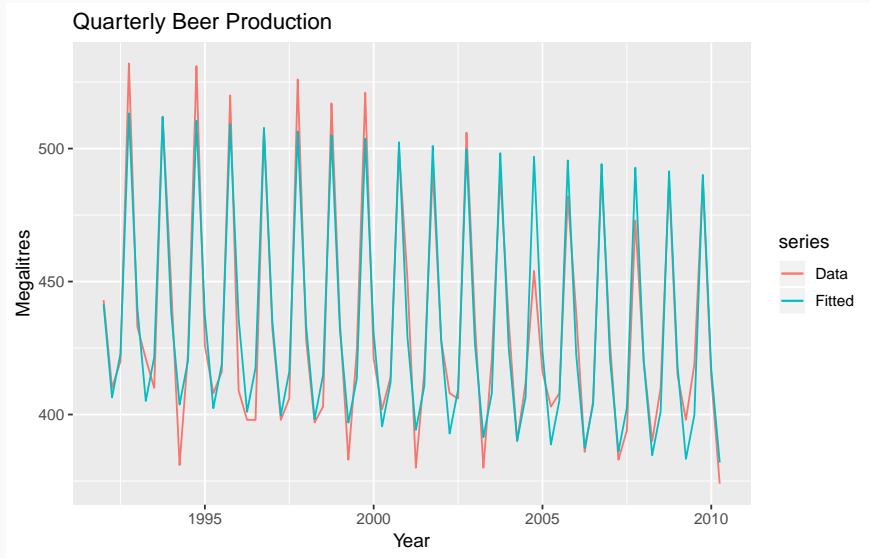
	Min	1Q	Median	3Q	Max
##	-42.903	-7.599	-0.459	7.991	21.789

```
##
## Coefficients:
```

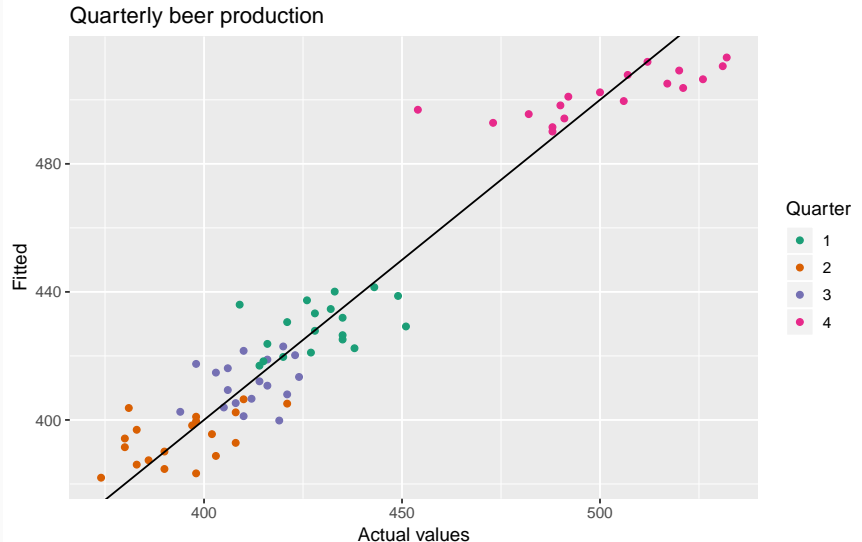
	Estimate	Std. Error	t value	Pr(> t )
## (Intercept)	441.80044	3.73353	118.333	< 2e-16 ***
## trend	-0.34027	0.06657	-5.111	2.73e-06 ***
## season2	-34.65973	3.96832	-8.734	9.10e-13 ***
## season3	-17.82164	4.02249	-4.430	3.45e-05 ***
## season4	72.79641	4.02305	18.095	< 2e-16 ***

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 12.23 on 69 degrees of freedom
## Multiple R-squared:  0.9243, Adjusted R-squared:  0.9199
## F-statistic: 210.7 on 4 and 69 DF,  p-value: < 2.2e-16
```

# Beer production revisited



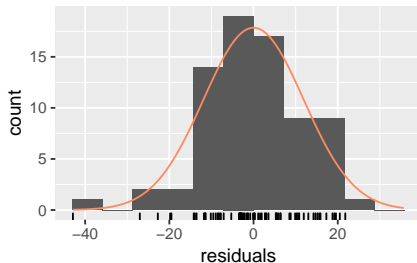
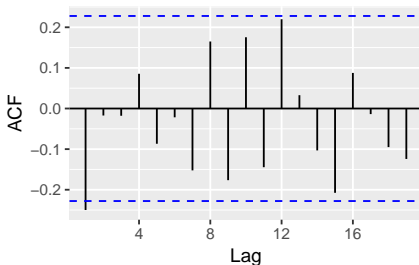
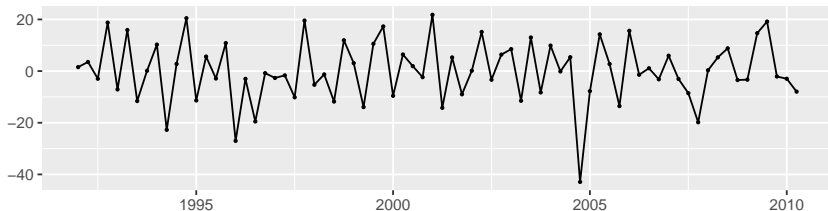
# Beer production revisited



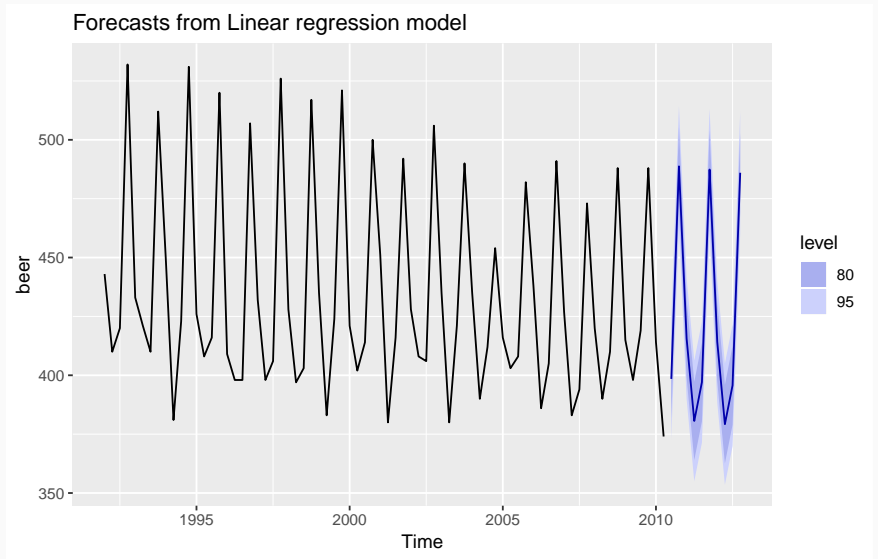


# Beer production revisited

Residuals from Linear regression model



# Beer production revisited



# 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”

```
fit <- tslm(y ~ trend + fourier(y, K))
```

# Harmonic regression: beer production

```
fourier.beer <- tslm(beer ~ trend + fourier(beer, K=2))
summary(fourier.beer)

##
## Call:
## tslm(formula = beer ~ trend + fourier(beer, K = 2))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -42.903  -7.599  -0.459   7.991  21.789
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    446.87920     2.87321 155.533 < 2e-16 ***
## trend          -0.34027     0.06657  -5.111 2.73e-06 ***
## fourier(beer, K = 2)S1-4    8.91082     2.01125   4.430 3.45e-05 ***
## fourier(beer, K = 2)C1-4   53.72807     2.01125  26.714 < 2e-16 ***
## fourier(beer, K = 2)C2-4   13.98958     1.42256   9.834 9.26e-15 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 12.23 on 69 degrees of freedom
## Multiple R-squared:  0.9243, Adjusted R-squared:  0.9199
## F-statistic: 210.7 on 4 and 69 DF,  p-value: < 2.2e-16
```

# 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  $\{1, 2, 3, \dots\}$  afterwards.

## 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.



# Trading days

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

$x_1$  = advertising for previous month;

$x_2$  = advertising for two months previously;

$\vdots$

$x_m$  = advertising for  $m$  months previously.

# Nonlinear trend

**Piecewise linear trend with bend at  $\tau$**

$$x_{1,t} = t$$

$$x_{2,t} = \begin{cases} 0 & t < \tau \\ (t - \tau) & t \geq \tau \end{cases}$$

# Nonlinear trend

**Piecewise linear trend with bend at  $\tau$**

$$x_{1,t} = t$$

$$x_{2,t} = \begin{cases} 0 & t < \tau \\ (t - \tau) & t \geq \tau \end{cases}$$

**Quadratic or higher order trend**

$$x_{1,t} = t, \quad x_{2,t} = t^2, \quad \dots$$

# Nonlinear trend

**Piecewise linear trend with bend at  $\tau$**

$$x_{1,t} = t$$

$$x_{2,t} = \begin{cases} 0 & t < \tau \\ (t - \tau) & t \geq \tau \end{cases}$$

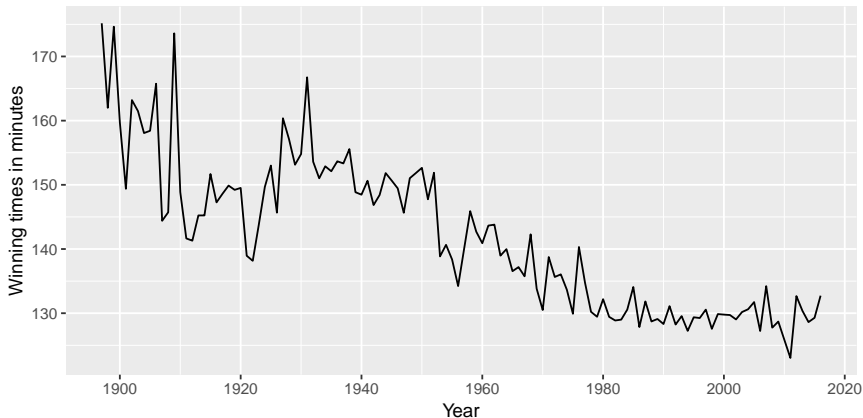
**Quadratic or higher order trend**

$$x_{1,t} = t, \quad x_{2,t} = t^2, \quad \dots$$

**NOT RECOMMENDED!**

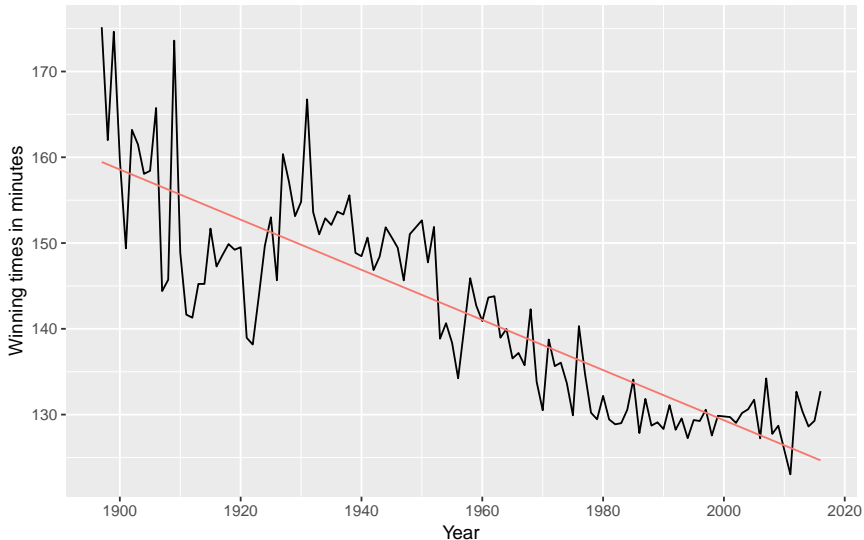
# Example: Boston marathon winning times

```
autoplot(marathon) +  
  xlab("Year") + ylab("Winning times in minutes")
```



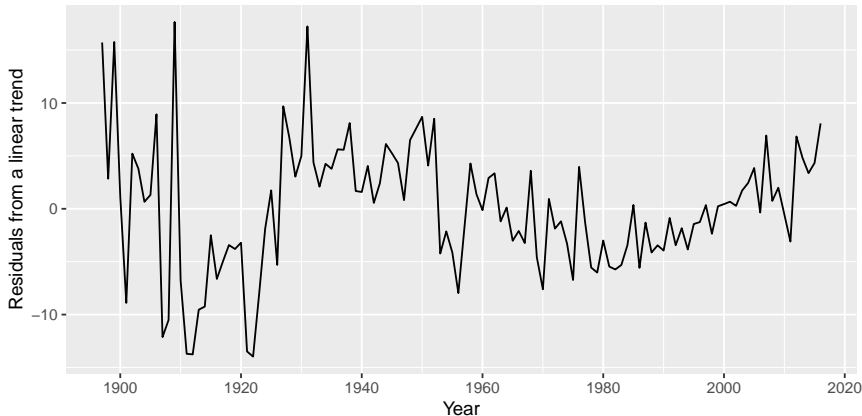
```
fit.lin <- tslm(marathon ~ trend)
```

# Example: Boston marathon winning times



# Example: Boston marathon winning times

```
autoplot(residuals(fit.lin)) +  
  xlab("Year") + ylab("Residuals from a linear trend")
```





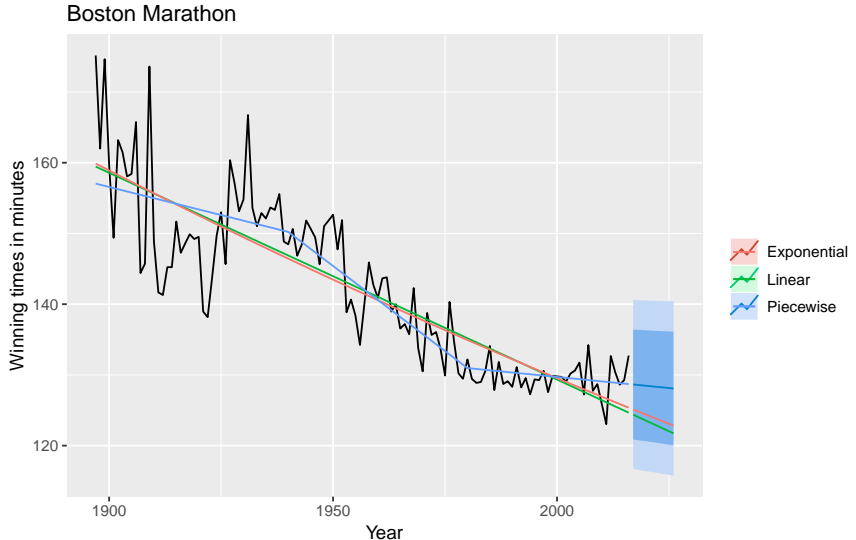
# Example: Boston marathon winning times

```
# Linear trend
fit.lin <- tslm(marathon ~ trend)
fcasts.lin <- forecast(fit.lin, h=10)

# Exponential trend
fit.exp <- tslm(marathon ~ trend, lambda = 0)
fcasts.exp <- forecast(fit.exp, h=10)

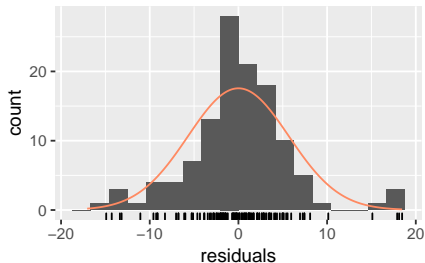
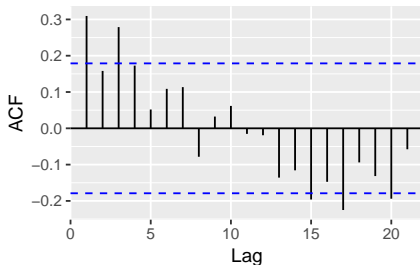
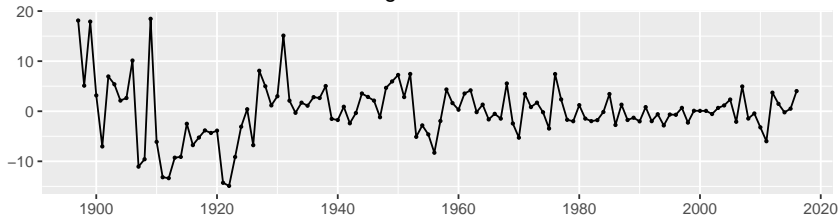
# Piecewise linear trend
t.break1 <- 1940
t.break2 <- 1980
t <- time(marathon)
t1 <- ts(pmax(0, t-t.break1), start=1897)
t2 <- ts(pmax(0, t-t.break2), start=1897)
fit.pw <- tslm(marathon ~ t + t1 + t2)
t.new <- t[length(t)] + seq(10)
t1.new <- t1[length(t1)] + seq(10)
t2.new <- t2[length(t2)] + seq(10)
newdata <- cbind(t=t.new, t1=t1.new, t2=t2.new) %>%
  as.data.frame
fcasts.pw <- forecast(fit.pw, newdata = newdata)
```

# Example: Boston marathon winning times

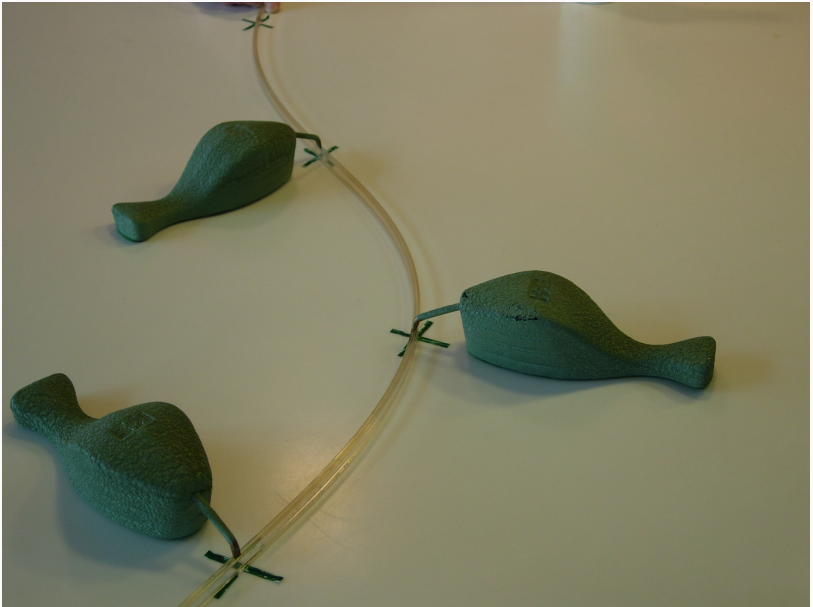


# Example: Boston marathon winning times

Residuals from Piecewise linear regression model



# Interpolating splines



# Interpolating splines



# Interpolating splines



# Interpolating splines

A spline is a continuous function  $f(x)$  interpolating all points  $(\kappa_j, y_j)$  for  $j = 1, \dots, K$  and consisting of polynomials between each consecutive pair of 'knots'  $\kappa_j$  and  $\kappa_{j+1}$ .

# Interpolating splines

A spline is a continuous function  $f(x)$  interpolating all points  $(\kappa_j, y_j)$  for  $j = 1, \dots, K$  and consisting of polynomials between each consecutive pair of 'knots'  $\kappa_j$  and  $\kappa_{j+1}$ .

- Parameters constrained so that  $f(x)$  is continuous.
- Further constraints imposed to give continuous derivatives.



# General linear regression splines

- Let  $\kappa_1 < \kappa_2 < \dots < \kappa_K$  be “knots” in interval  $(a, b)$ .
- Let  $x_1 = x$ ,  $x_j = (x - \kappa_{j-1})_+$  for  $j = 2, \dots, K + 1$ .
- Then the regression is piecewise linear with bends at the knots.

# General cubic splines

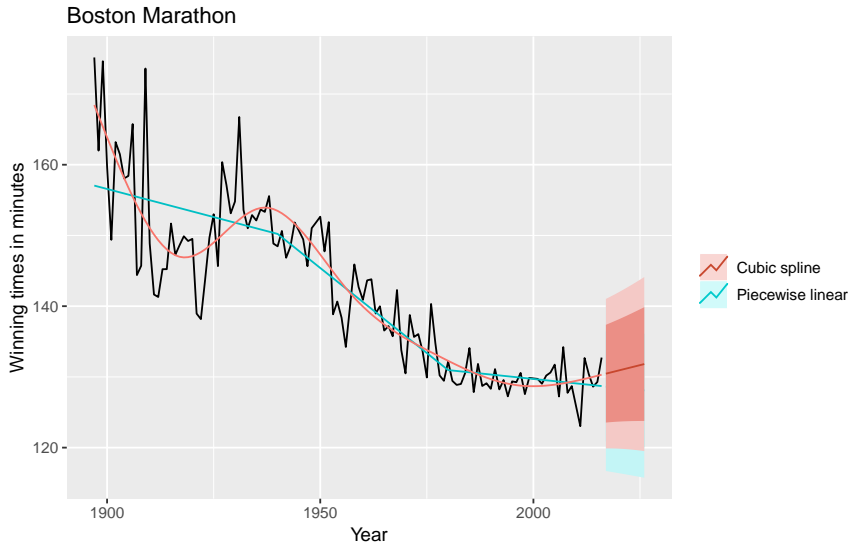
- Let  $x_1 = x$ ,  $x_2 = x^2$ ,  $x_3 = x^3$ ,  $x_j = (x - \kappa_{j-3})_+^3$  for  $j = 4, \dots, K + 3$ .
- Then the regression is piecewise cubic, but smooth at the knots.
- Choice of knots can be difficult and arbitrary.
- Automatic knot selection algorithms very slow.

# Example: Boston marathon winning times

```
# Spline trend
library(splines)
t <- time(marathon)
fit.splines <- lm(marathon ~ ns(t, df=6))
summary(fit.splines)

##
## Call:
## lm(formula = marathon ~ ns(t, df = 6))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -13.0028  -2.5722   0.0122   2.1242  21.5681
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    168.447     2.086  80.743  < 2e-16 ***
## ns(t, df = 6)1    -6.948     2.688  -2.584   0.011 *
## ns(t, df = 6)2   -28.856     3.416  -8.448 1.16e-13 ***
## ns(t, df = 6)3   -35.081     3.045 -11.522  < 2e-16 ***
## ns(t, df = 6)4   -32.563     2.652 -12.279  < 2e-16 ***
## ns(t, df = 6)5   -64.847     5.322 -12.184  < 2e-16 ***
## ns(t, df = 6)6   -21.002     2.403  -8.741 2.46e-14 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.834 on 113 degrees of freedom
## Multiple R-squared:  0.8418, Adjusted R-squared:  0.8334
## F-statistic: 100.2 on 6 and 113 DF, p-value: < 2.2e-16
```

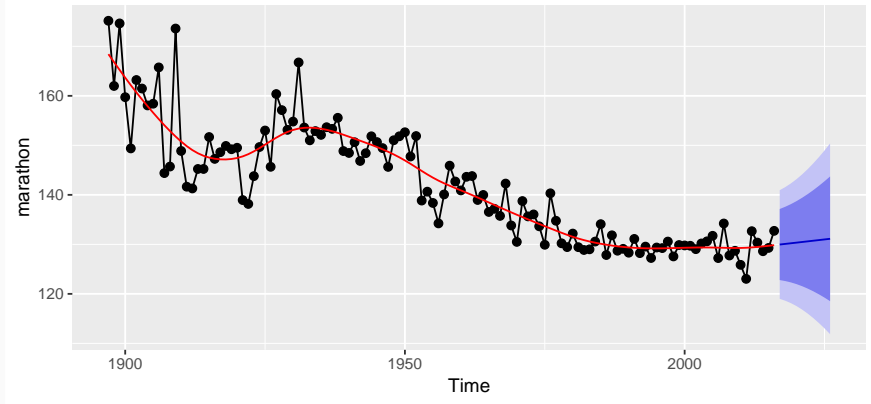
# Example: Boston marathon winning times



# splinef

A slightly different type of spline is provided by `splinef`

```
fc <- splinef(marathon)  
autoplot(fc)
```



- Cubic **smoothing** splines (rather than cubic regression splines).
- Still piecewise cubic, but with many more knots (one at each observation).
- Coefficients constrained to prevent the curve becoming too “wiggly”.
- Degrees of freedom selected automatically.
- Equivalent to ARIMA(0,2,2) and Holt’s method.

# Outline

- 1 The linear model with time series
- 2 Residual diagnostics
- 3 Some useful predictors for linear models
- 4 Selecting predictors and forecast evaluation**
- 5 Forecasting with regression
- 6 Matrix formulation
- 7 Correlation, causation and forecasting

# Selecting predictors

- When there are many predictors, how should we choose which ones to use?
- We need a way of comparing two competing models.



# Selecting predictors

- When there are many predictors, how should we choose which ones to use?
- We need a way of comparing two competing models.

## What not to do!

- Plot  $y$  against a particular predictor ( $x_j$ ) and if it shows no noticeable relationship, drop it.
- Do a multiple linear regression on all the predictors and disregard all variables whose  $p$  values are greater than 0.05.
- Maximize  $R^2$  or minimize MSE

# Comparing regression models

Computer output for regression will always give the  $R^2$  value. This is a useful summary of the model.

- It is equal to the square of the correlation between  $y$  and  $\hat{y}$ .
- It is often called the “coefficient of determination”.
- It can also be calculated as follows:

$$R^2 = \frac{\sum(\hat{y}_t - \bar{y})^2}{\sum(y_t - \bar{y})^2}$$

- It is the proportion of variance accounted for (explained) by the predictors.

# Comparing regression models

However ...

- $R^2$  does not allow for “degrees of freedom”.
- Adding *any* variable tends to increase the value of  $R^2$ , even if that variable is irrelevant.

# Comparing regression models

However ...

- $R^2$  does not allow for “degrees of freedom”.
- Adding *any* variable tends to increase the value of  $R^2$ , even if that variable is irrelevant.

To overcome this problem, we can use *adjusted*  $R^2$ :

$$\bar{R}^2 = 1 - (1 - R^2) \frac{T - 1}{T - k - 1}$$

where  $k$  = no. predictors and  $T$  = no. observations.

# Comparing regression models

However ...

- $R^2$  does not allow for “degrees of freedom”.
- Adding *any* variable tends to increase the value of  $R^2$ , even if that variable is irrelevant.

To overcome this problem, we can use *adjusted*  $R^2$ :

$$\bar{R}^2 = 1 - (1 - R^2) \frac{T - 1}{T - k - 1}$$

where  $k$  = no. predictors and  $T$  = no. observations.

**Maximizing  $\bar{R}^2$  is equivalent to minimizing  $\hat{\sigma}^2$ .**

$$\hat{\sigma}^2 = \frac{1}{T - k - 1} \sum_{t=1}^T \varepsilon_t^2$$

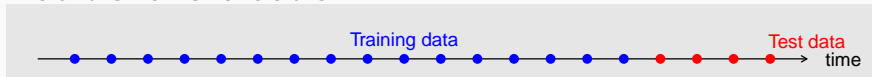
## Cross-validation for regression

(Assuming future predictors are known)

- Select one observation for test set, and use *remaining* observations in training set. Compute error on test observation.
- Repeat using each possible observation as the test set.
- Compute accuracy measure over all errors.

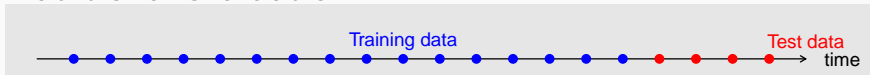
# Cross-validation

## Traditional evaluation

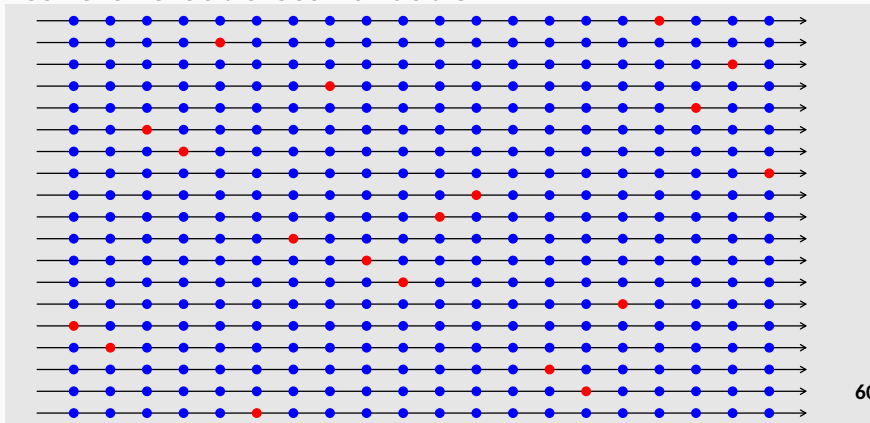


# Cross-validation

## Traditional evaluation



## Leave-one-out cross-validation





# Cross-validation

Leave-one-out cross-validation for regression can be carried out using the following steps.

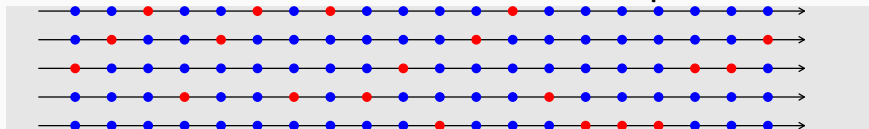
- Remove observation  $t$  from the data set, and fit the model using the remaining data. Then compute the error ( $e_t^* = y_t - \hat{y}_t$ ) for the omitted observation.
- Repeat step 1 for  $t = 1, \dots, T$ .
- Compute the MSE from  $\{e_1^*, \dots, e_T^*\}$ . We shall call this the CV.

The best model is the one with minimum CV.

# Cross-validation

## Five-fold cross-validation

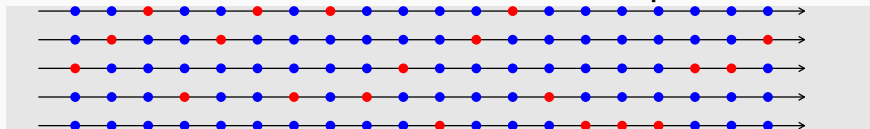
- 20 observations. 4 test observations per fold



# Cross-validation

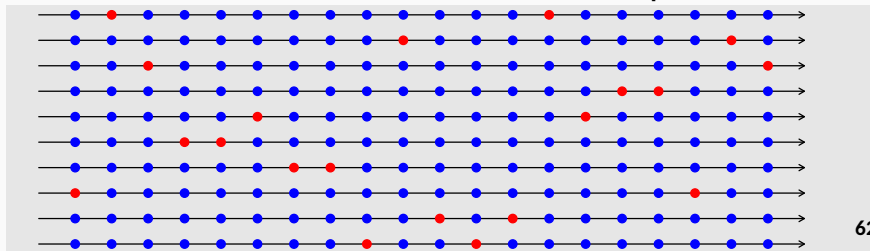
## Five-fold cross-validation

- 20 observations. 4 test observations per fold



## Ten-fold cross-validation

- 20 observations. 2 test observations per fold



## Ten-fold cross-validation

- Randomly split data into 10 parts.
- Select one part for test set, and use remaining parts as training set. Compute accuracy measures on test observations.
- Repeat for each of 10 parts
- Average over all measures.

# Akaike's Information Criterion

$$\text{AIC} = -2 \log(L) + 2(k + 2)$$

where  $L$  is the likelihood and  $k$  is the number of predictors in the model.

# Akaike's Information Criterion

$$\text{AIC} = -2 \log(L) + 2(k + 2)$$

where  $L$  is the likelihood and  $k$  is the number of predictors in the model.

- This is a *penalized likelihood* approach.
- *Minimizing the AIC gives the best model for prediction.*
- AIC penalizes terms more heavily than  $\bar{R}^2$ .
- Minimizing the AIC is asymptotically equivalent to minimizing MSE via leave-one-out cross-validation.

## Corrected AIC

For small values of  $T$ , the AIC tends to select too many predictors, and so a bias-corrected version of the AIC has been developed.

$$\text{AIC}_C = \text{AIC} + \frac{2(k+2)(k+3)}{T-k-3}$$

As with the AIC, the  $\text{AIC}_C$  should be minimized.

# Bayesian Information Criterion

$$\text{BIC} = -2 \log(L) + (k + 2) \log(T)$$

where  $L$  is the likelihood and  $k$  is the number of predictors in the model.



# Bayesian Information Criterion

$$\text{BIC} = -2 \log(L) + (k + 2) \log(T)$$

where  $L$  is the likelihood and  $k$  is the number of predictors in the model.

- BIC penalizes terms more heavily than AIC
- Also called SBIC and SC.
- Minimizing BIC is asymptotically equivalent to leave- $v$ -out cross-validation when  $v = T[1 - 1/(\log(T) - 1)]$ .

# Choosing regression variables

## Best subsets regression

- Fit all possible regression models using one or more of the predictors.
- Choose the best model based on one of the measures of predictive ability (CV, AIC, AICc).

# Choosing regression variables

## Best subsets regression

- Fit all possible regression models using one or more of the predictors.
- Choose the best model based on one of the measures of predictive ability (CV, AIC, AICc).

## Warning!

- If there are a large number of predictors, this is not possible.
- For example, 44 predictors leads to 18 trillion possible models!

# Choosing regression variables

## Backwards stepwise regression

- Start with a model containing all variables.
- Try subtracting one variable at a time. Keep the model if it has lower CV or AICc.
- Iterate until no further improvement.

# Choosing regression variables

## Backwards stepwise regression

- Start with a model containing all variables.
- Try subtracting one variable at a time. Keep the model if it has lower CV or AICc.
- Iterate until no further improvement.

## Notes

- Stepwise regression is not guaranteed to lead to the best possible model.
- Inference on coefficients of final model will be wrong.

# Cross-validation



```
tslm(Consumption ~ Income + Production + Unemployment + Savings,  
data=uschange) %>% CV()
```

##	CV	AIC	AICc	BIC	AdjR2
##	0.1163477	-409.2980298	-408.8313631	-389.9113781	0.7485856

```
tslm(Consumption ~ Income + Production + Unemployment,  
data=uschange) %>% CV()
```

##	CV	AIC	AICc	BIC	AdjR2
##	0.2776928	-243.1635677	-242.8320760	-227.0080246	0.3855438

```
tslm(Consumption ~ Income + Production + Savings,  
data=uschange) %>% CV()
```

##	CV	AIC	AICc	BIC	AdjR2
##	0.1178681	-407.4669279	-407.1354362	-391.3113848	0.7447840

```
tslm(Consumption ~ Income + Unemployment + Savings,  
data=uschange) %>% CV()
```

##	CV	AIC	AICc	BIC	AdjR2
##	0.1160223	-408.0941325	-407.7626408	-391.9385894	0.7456386

```
tslm(Consumption ~ Production + Unemployment + Savings,  
data=uschange) %>% CV()
```

##	CV	AIC	AICc	BIC	AdjR2
##	0.2927095	-234.3734580	-234.0419663	-218.2179149	0.3559711

# Outline

- 1 The linear model with time series
- 2 Residual diagnostics
- 3 Some useful predictors for linear models
- 4 Selecting predictors and forecast evaluation
- 5 Forecasting with regression**
- 6 Matrix formulation
- 7 Correlation, causation and forecasting

# Ex-ante versus ex-post forecasts

- *Ex ante forecasts* are made using only information available in advance.
  - require forecasts of predictors
- *Ex post forecasts* are made using later information on the predictors.
  - useful for studying behaviour of forecasting models.
- trend, seasonal and calendar variables are all known in advance, so these don't need to be forecast.



# Scenario based forecasting

- Assumes possible scenarios for the predictor variables
- Prediction intervals for scenario based forecasts do not include the uncertainty associated with the future values of the predictor variables.

# Building a predictive regression model

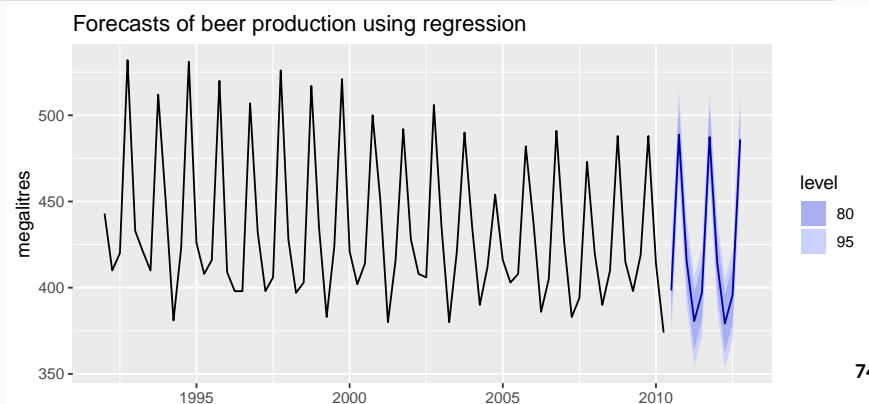
- If getting forecasts of predictors is difficult, you can use lagged predictors instead.

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

- A different model for each forecast horizon  $h$ .

# Beer production

```
beer2 <- window(ausbeer, start=1992)
fit.beer <- tslm(beer2 ~ trend + season)
fcast <- forecast(fit.beer)
autoplot(fcast) +
  ggtitle("Forecasts of beer production using regression") +
  xlab("Year") + ylab("megalitres")
```

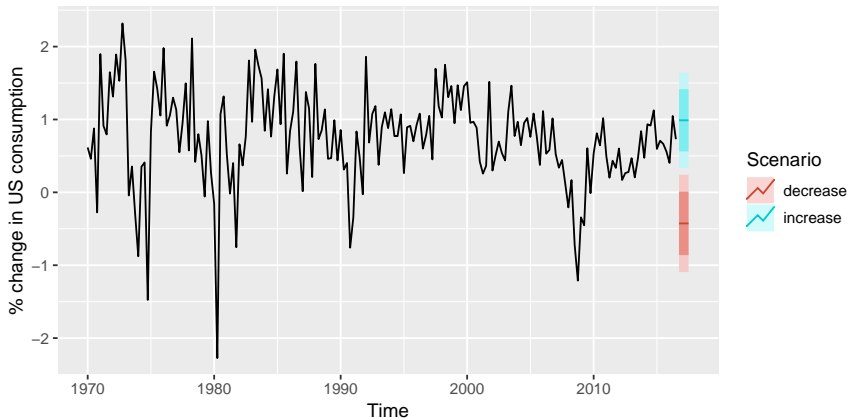


# US Consumption

```
fit.consBest <- tslm(  
  Consumption ~ Income + Savings + Unemployment,  
  data = uschange)  
h <- 4  
newdata <- data.frame(  
  Income = c(1, 1, 1, 1),  
  Savings = c(0.5, 0.5, 0.5, 0.5),  
  Unemployment = c(0, 0, 0, 0))  
fcast.up <- forecast(fit.consBest, newdata = newdata)  
newdata <- data.frame(  
  Income = rep(-1, h),  
  Savings = rep(-0.5, h),  
  Unemployment = rep(0, h))  
fcast.down <- forecast(fit.consBest, newdata = newdata)
```

# US Consumption

```
autoplot(uschange[, 1]) +  
  ylab("% change in US consumption") +  
  autolayer(fcast.up, PI = TRUE, series = "increase") +  
  autolayer(fcast.down, PI = TRUE, series = "decrease") +  
  guides(colour = guide_legend(title = "Scenario"))
```



# Outline

- 1 The linear model with time series
- 2 Residual diagnostics
- 3 Some useful predictors for linear models
- 4 Selecting predictors and forecast evaluation
- 5 Forecasting with regression
- 6 Matrix formulation**
- 7 Correlation, causation and forecasting

# Matrix formulation

$$y_t = \beta_0 + \beta_1 x_{1,t} + \beta_2 x_{2,t} + \cdots + \beta_k x_{k,t} + \varepsilon_t.$$

# Matrix formulation

$$y_t = \beta_0 + \beta_1 x_{1,t} + \beta_2 x_{2,t} + \dots + \beta_k x_{k,t} + \varepsilon_t.$$

Let  $\mathbf{y} = (y_1, \dots, y_T)'$ ,  $\boldsymbol{\varepsilon} = (\varepsilon_1, \dots, \varepsilon_T)'$ ,

$\boldsymbol{\beta} = (\beta_0, \beta_1, \dots, \beta_k)'$  and

$$\mathbf{X} = \begin{bmatrix} 1 & x_{1,1} & x_{2,1} & \dots & x_{k,1} \\ 1 & x_{1,2} & x_{2,2} & \dots & x_{k,2} \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & x_{1,T} & x_{2,T} & \dots & x_{k,T} \end{bmatrix}.$$



# Matrix formulation

$$y_t = \beta_0 + \beta_1 x_{1,t} + \beta_2 x_{2,t} + \dots + \beta_k x_{k,t} + \varepsilon_t.$$

Let  $\mathbf{y} = (y_1, \dots, y_T)'$ ,  $\boldsymbol{\varepsilon} = (\varepsilon_1, \dots, \varepsilon_T)'$ ,

$\boldsymbol{\beta} = (\beta_0, \beta_1, \dots, \beta_k)'$  and

$$\mathbf{X} = \begin{bmatrix} 1 & x_{1,1} & x_{2,1} & \dots & x_{k,1} \\ 1 & x_{1,2} & x_{2,2} & \dots & x_{k,2} \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & x_{1,T} & x_{2,T} & \dots & x_{k,T} \end{bmatrix}.$$

Then

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon}.$$

# Matrix formulation

## Least squares estimation

Minimize:  $(\mathbf{y} - \mathbf{X}\beta)'(\mathbf{y} - \mathbf{X}\beta)$

# Matrix formulation

## Least squares estimation

Minimize:  $(\mathbf{y} - \mathbf{X}\beta)'(\mathbf{y} - \mathbf{X}\beta)$

Differentiate wrt  $\beta$  gives

$$\hat{\beta} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{y}$$

# Matrix formulation

## Least squares estimation

Minimize:  $(\mathbf{y} - \mathbf{X}\beta)'(\mathbf{y} - \mathbf{X}\beta)$

Differentiate wrt  $\beta$  gives

$$\hat{\beta} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{y}$$

(The “normal equation”.)

# Matrix formulation

## Least squares estimation

Minimize:  $(\mathbf{y} - \mathbf{X}\beta)'(\mathbf{y} - \mathbf{X}\beta)$

Differentiate wrt  $\beta$  gives

$$\hat{\beta} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{y}$$

(The “normal equation”.)

$$\hat{\sigma}^2 = \frac{1}{T - k - 1}(\mathbf{y} - \mathbf{X}\hat{\beta})'(\mathbf{y} - \mathbf{X}\hat{\beta})$$

**Note:** If you fall for the dummy variable trap,  $(\mathbf{X}'\mathbf{X})$  is a singular matrix.

If the errors are iid and normally distributed, then

$$\mathbf{y} \sim N(\mathbf{X}\boldsymbol{\beta}, \sigma^2 \mathbf{I}).$$

# Likelihood

If the errors are iid and normally distributed, then

$$\mathbf{y} \sim N(\mathbf{X}\boldsymbol{\beta}, \sigma^2 \mathbf{I}).$$

So the likelihood is

$$L = \frac{1}{\sigma^T (2\pi)^{T/2}} \exp \left( -\frac{1}{2\sigma^2} (\mathbf{y} - \mathbf{X}\boldsymbol{\beta})' (\mathbf{y} - \mathbf{X}\boldsymbol{\beta}) \right)$$

# Likelihood

If the errors are iid and normally distributed, then

$$\mathbf{y} \sim N(\mathbf{X}\beta, \sigma^2 \mathbf{I}).$$

So the likelihood is

$$L = \frac{1}{\sigma^T (2\pi)^{T/2}} \exp \left( -\frac{1}{2\sigma^2} (\mathbf{y} - \mathbf{X}\beta)' (\mathbf{y} - \mathbf{X}\beta) \right)$$

which is maximized when  $(\mathbf{y} - \mathbf{X}\beta)' (\mathbf{y} - \mathbf{X}\beta)$  is minimized.



# Likelihood

If the errors are iid and normally distributed, then

$$\mathbf{y} \sim N(\mathbf{X}\boldsymbol{\beta}, \sigma^2 \mathbf{I}).$$

So the likelihood is

$$L = \frac{1}{\sigma^T (2\pi)^{T/2}} \exp \left( -\frac{1}{2\sigma^2} (\mathbf{y} - \mathbf{X}\boldsymbol{\beta})' (\mathbf{y} - \mathbf{X}\boldsymbol{\beta}) \right)$$

which is maximized when  $(\mathbf{y} - \mathbf{X}\boldsymbol{\beta})' (\mathbf{y} - \mathbf{X}\boldsymbol{\beta})$  is minimized.

**So MLE = OLS.**

# Multiple regression forecasts

## Optimal forecasts

$$\hat{y}^* = E(y^* | \mathbf{y}, \mathbf{X}, \mathbf{x}^*) = \mathbf{x}^* \hat{\boldsymbol{\beta}} = \mathbf{x}^* (\mathbf{X}'\mathbf{X})^{-1} \mathbf{X}'\mathbf{y}$$

where  $\mathbf{x}^*$  is a row vector containing the values of the predictors for the forecasts (in the same format as  $\mathbf{X}$ ).

# Multiple regression forecasts

## Optimal forecasts

$$\hat{y}^* = E(y^* | \mathbf{y}, \mathbf{X}, \mathbf{x}^*) = \mathbf{x}^* \hat{\boldsymbol{\beta}} = \mathbf{x}^* (\mathbf{X}'\mathbf{X})^{-1} \mathbf{X}'\mathbf{y}$$

where  $\mathbf{x}^*$  is a row vector containing the values of the predictors for the forecasts (in the same format as  $\mathbf{X}$ ).

## Forecast variance

$$\text{Var}(y^* | \mathbf{X}, \mathbf{x}^*) = \sigma^2 \left[ 1 + \mathbf{x}^* (\mathbf{X}'\mathbf{X})^{-1} (\mathbf{x}^*)' \right]$$

# Multiple regression forecasts

## Optimal forecasts

$$\hat{y}^* = E(y^* | \mathbf{y}, \mathbf{X}, \mathbf{x}^*) = \mathbf{x}^* \hat{\boldsymbol{\beta}} = \mathbf{x}^* (\mathbf{X}'\mathbf{X})^{-1} \mathbf{X}'\mathbf{y}$$

where  $\mathbf{x}^*$  is a row vector containing the values of the predictors for the forecasts (in the same format as  $\mathbf{X}$ ).

## Forecast variance

$$\text{Var}(y^* | \mathbf{X}, \mathbf{x}^*) = \sigma^2 \left[ 1 + \mathbf{x}^* (\mathbf{X}'\mathbf{X})^{-1} (\mathbf{x}^*)' \right]$$

- This ignores any errors in  $\mathbf{x}^*$ .
- 95% prediction intervals assuming normal errors:

$$\hat{y}^* \pm 1.96 \sqrt{\text{Var}(y^* | \mathbf{X}, \mathbf{x}^*)}.$$

# Multiple regression forecasts

## Fitted values

$$\hat{\mathbf{y}} = \mathbf{X}\hat{\boldsymbol{\beta}} = \mathbf{X}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{y} = \mathbf{H}\mathbf{y}$$

where  $\mathbf{H} = \mathbf{X}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'$  is the “hat matrix”.

# Multiple regression forecasts

## Fitted values

$$\hat{\mathbf{y}} = \mathbf{X}\hat{\boldsymbol{\beta}} = \mathbf{X}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{y} = \mathbf{H}\mathbf{y}$$

where  $\mathbf{H} = \mathbf{X}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'$  is the “hat matrix”.

## Leave-one-out residuals

Let  $h_1, \dots, h_T$  be the diagonal values of  $\mathbf{H}$ , then the cross-validation statistic is

$$\text{CV} = \frac{1}{T} \sum_{t=1}^T [e_t / (1 - h_t)]^2,$$

where  $e_t$  is the residual obtained from fitting the model to all  $T$  observations.

# Outline

- 1 The linear model with time series
- 2 Residual diagnostics
- 3 Some useful predictors for linear models
- 4 Selecting predictors and forecast evaluation
- 5 Forecasting with regression
- 6 Matrix formulation
- 7 Correlation, causation and forecasting

# Correlation is not causation

- When  $x$  is useful for predicting  $y$ , it is not necessarily causing  $y$ .
- e.g., predict number of drownings  $y$  using number of ice-creams sold  $x$ .
- Correlations are useful for forecasting, even when there is no causality.
- Better models usually involve causal relationships (e.g., temperature  $x$  and people  $z$  to predict drownings  $y$ ).



# Multicollinearity

In regression analysis, multicollinearity occurs when:

- Two predictors are highly correlated (i.e., the correlation between them is close to  $\pm 1$ ).
- A linear combination of some of the predictors is highly correlated with another predictor.
- A linear combination of one subset of predictors is highly correlated with a linear combination of another subset of predictors.

# Multicollinearity

If multicollinearity exists...

- the numerical estimates of coefficients may be wrong (worse in Excel than in a statistics package)
- don't rely on the  $p$ -values to determine significance.
- there is no problem with model *predictions* provided the predictors used for forecasting are within the range used for fitting.
- omitting variables can help.
- combining variables can help.

# Outliers and influential observations

## Things to watch for

- *Outliers*: observations that produce large residuals.
- *Influential observations*: removing them would markedly change the coefficients. (Often outliers in the  $x$  variable).
- *Lurking variable*: a predictor not included in the regression but which has an important effect on the response.
- Points should not normally be removed without a good explanation of why they are different.