Applied Economic Forecasting

4. Time Series Regressions

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. Correlation, causation and forecasting
7. Nonlinear Regressions

Section 1

The linear model with time series

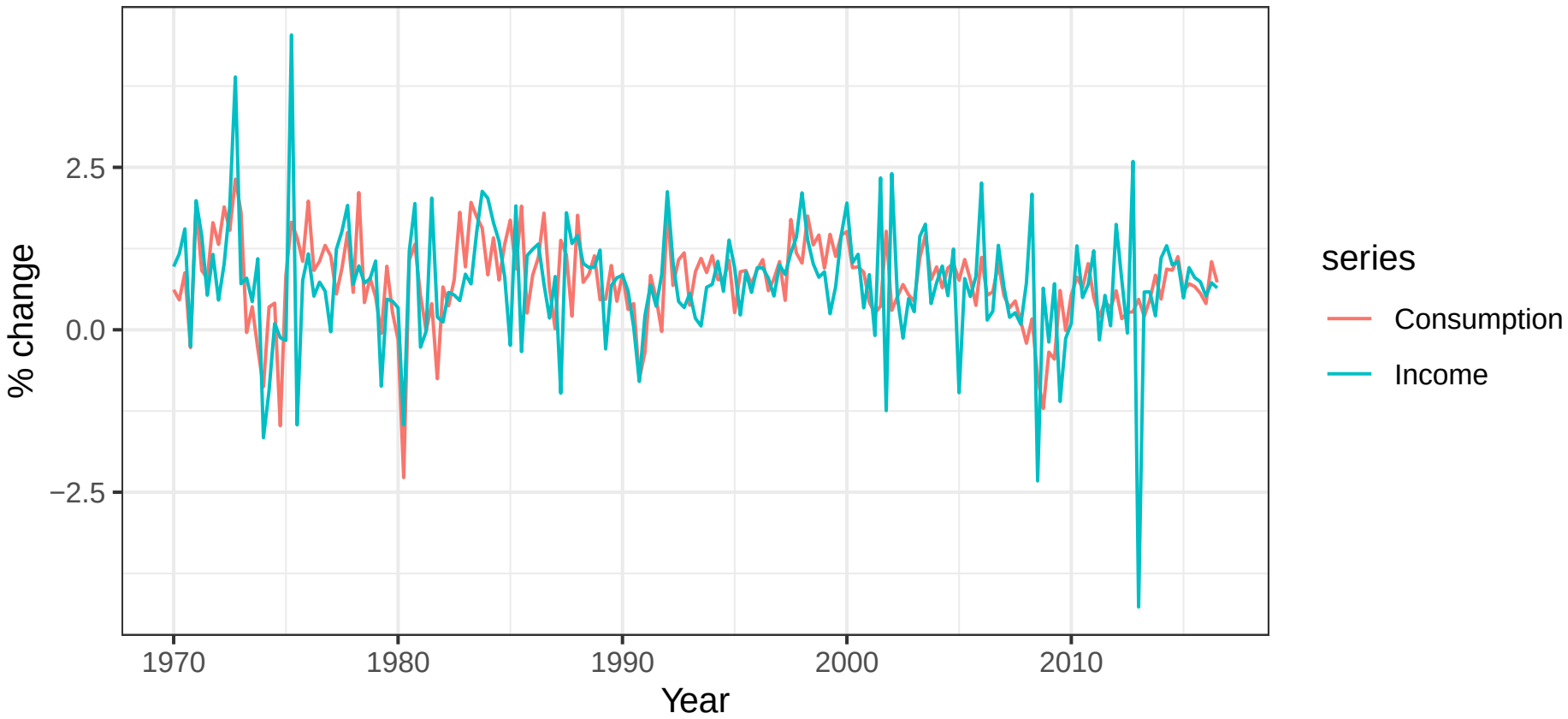
Multiple regression and forecasting

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

- 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 $\beta_1, \ldots, \beta_k$ 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**.
- ε_t is a white noise error term

Example: US consumption expenditure

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



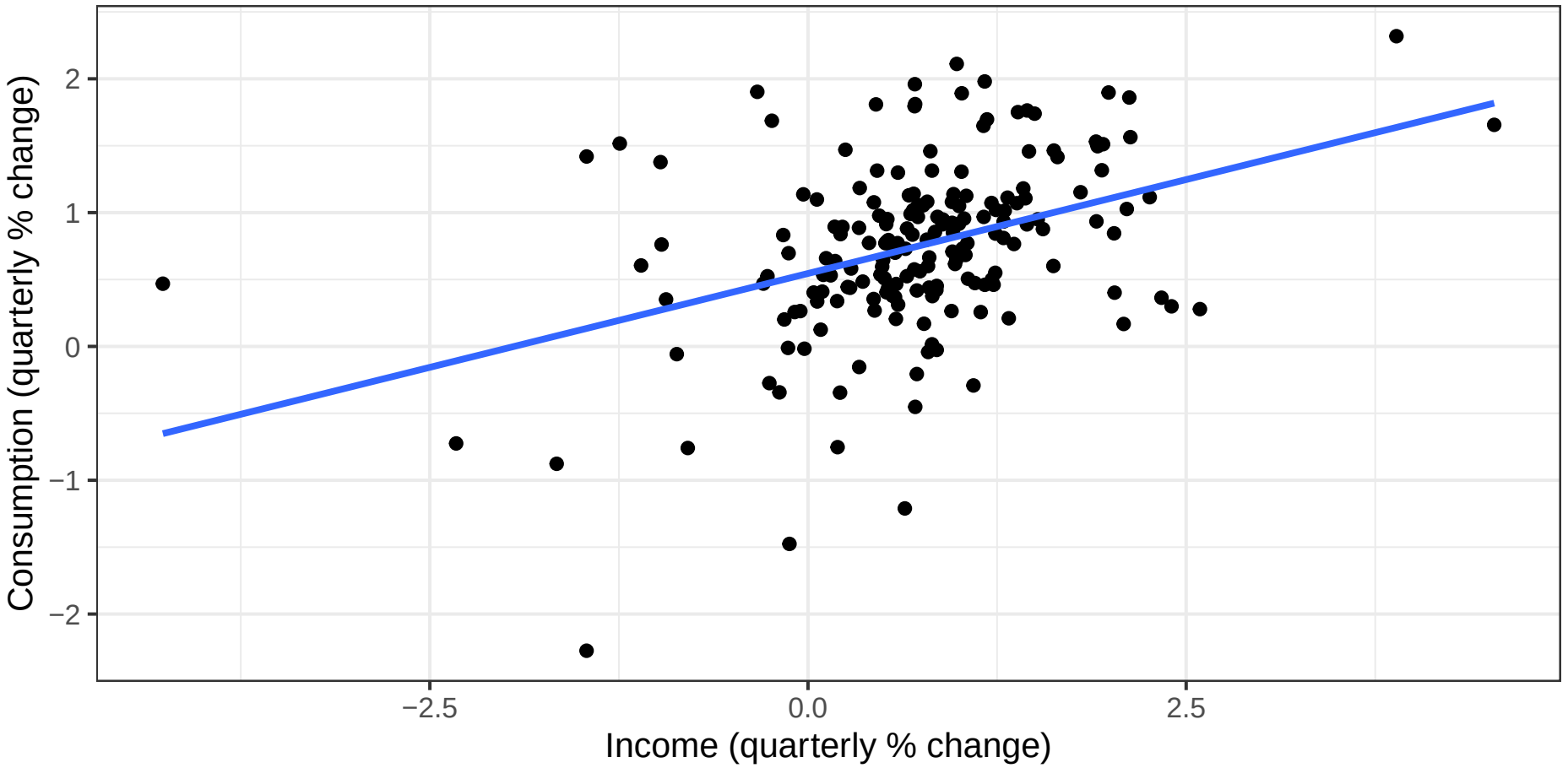
Example: US consumption expenditure

```
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
```

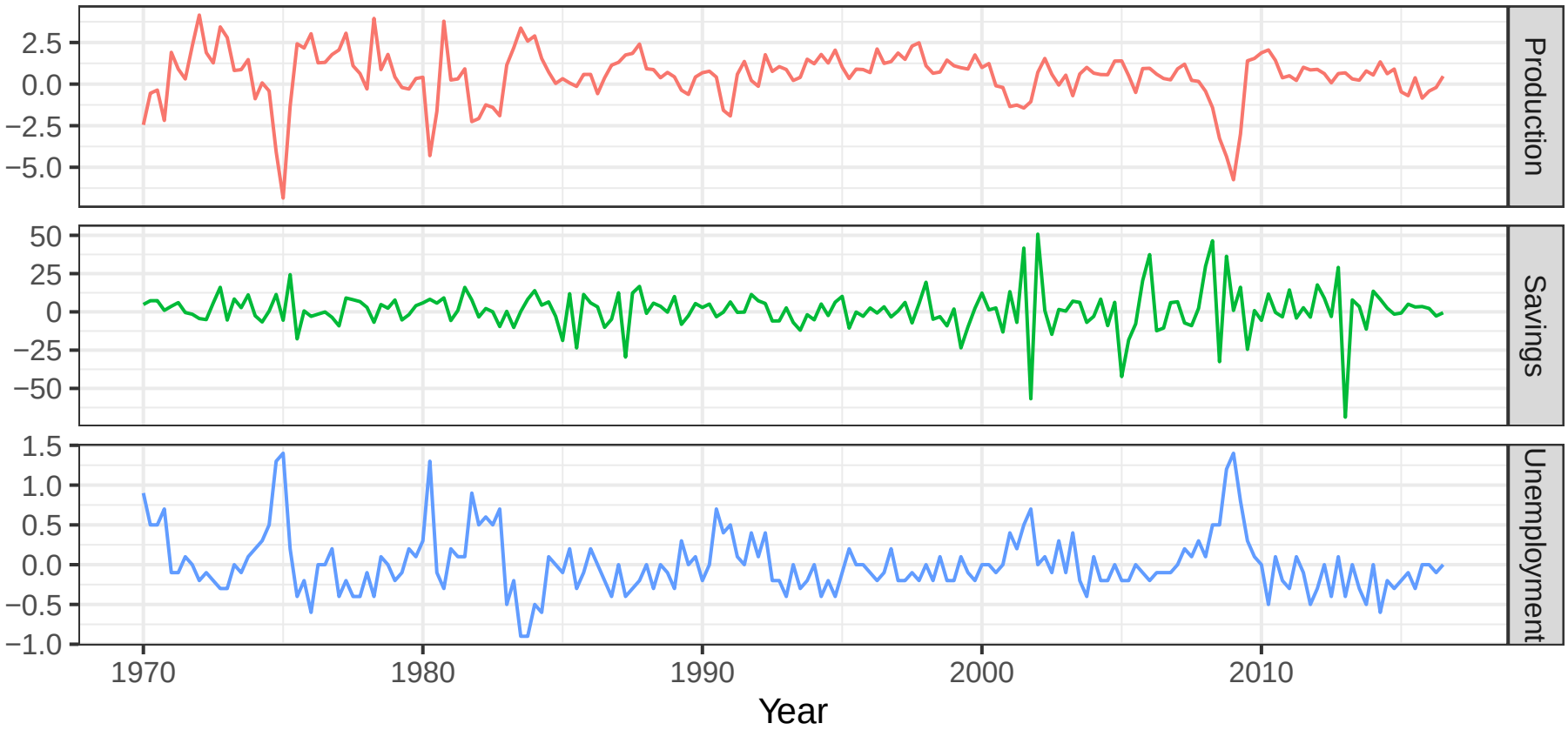
Example: US consumption expenditure

```
fit.cons <- tslm(Consumption ~ Income, data=uschange)
uschange %>%  as.data.frame %>%  ggplot(aes(x=Income, y=Consumption)) +
  geom_point() + geom_smooth(method="lm", se=FALSE) +
  labs(y="Consumption (quarterly % change)", x = "Income (quarterly % change)")
```



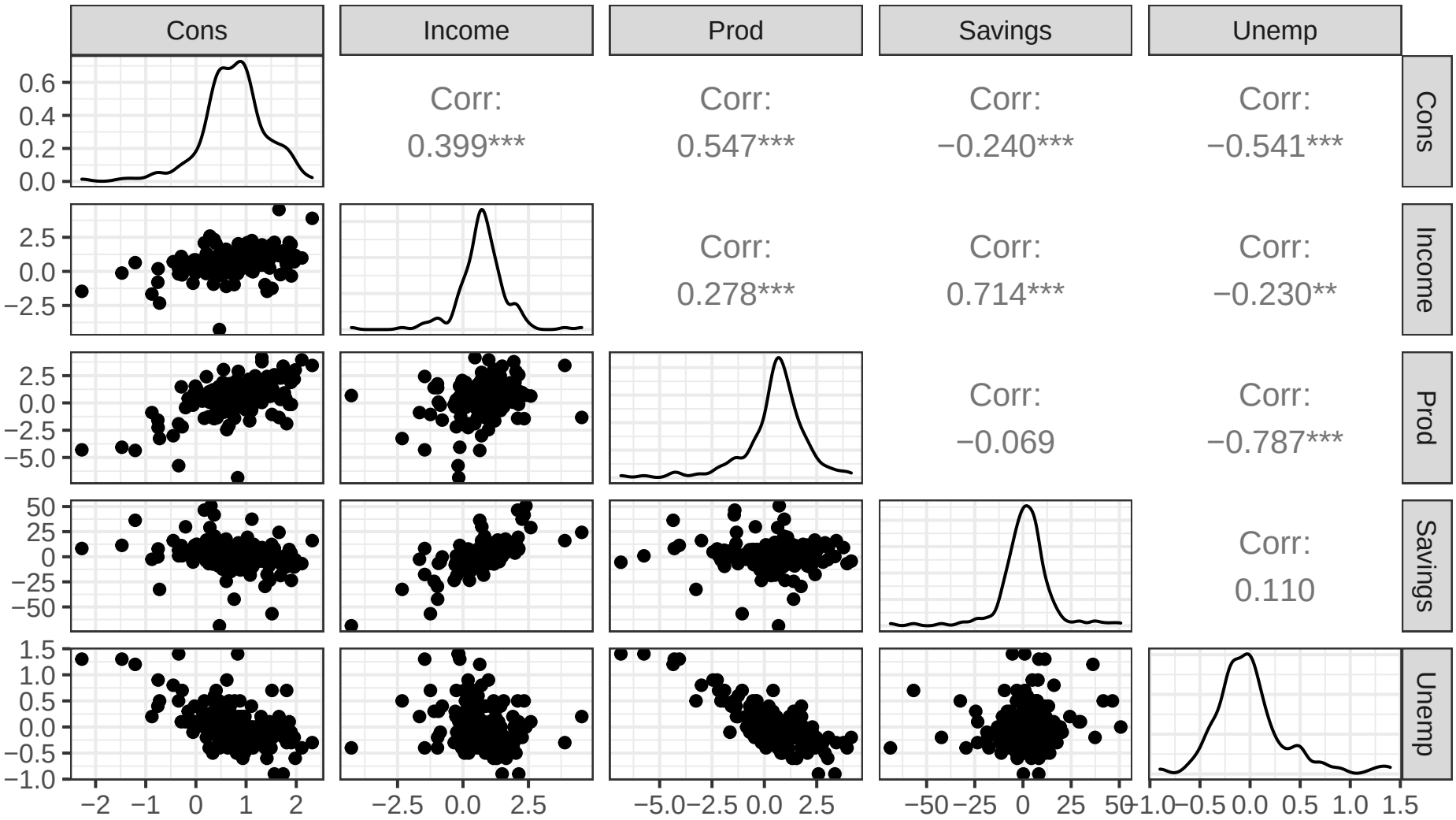
Example: US consumption expenditure

```
autoplot(uschange[,3:5], facets = TRUE, colour=TRUE) +
  labs(y = "", x = "Year") +  guides(colour="none")
```



Example: US consumption expenditure

```
uschange %>% as.data.frame %>% GGally::ggpairs(columnLabels = c("Cons", "Income",
    "Prod", "Savings", "Unemp"))
```



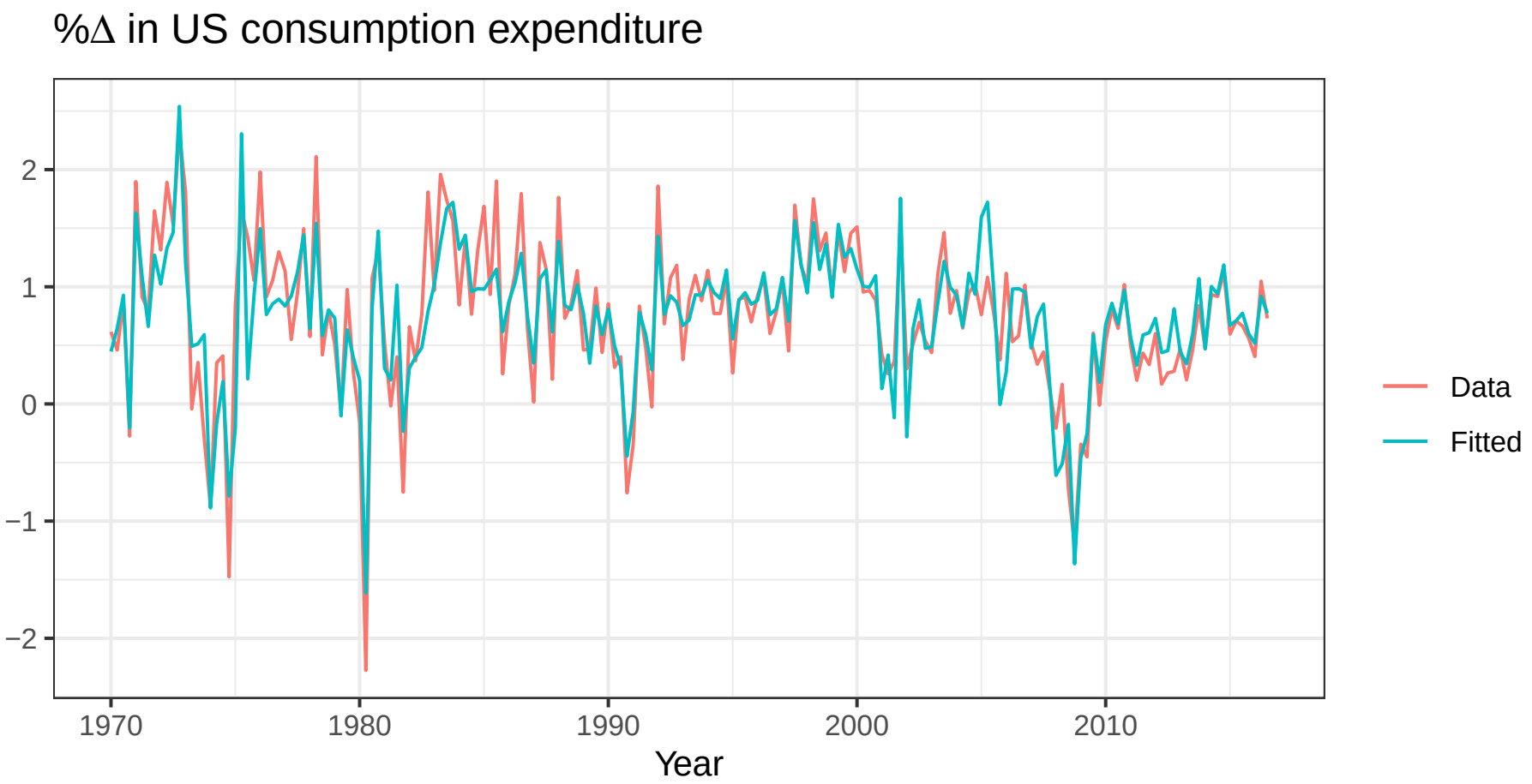
Example: US consumption expenditure

```
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

```
autoplot(uschange[, "Consumption"], series = "Data") +
    autolayer(fitted(fit.consMR), series = "Fitted") +
    labs(x = "Year", y = "") + ggtitle(TeX("%$\\Delta$ in US consumption expenditure
    guides(colour = guide_legend(title = " "))
```



Goodness of Fit: R^2

A common way to summarise how well a linear regression model fits the data is via the coefficient of determination, or R^2. R^2 tells us how much of the variation in our dependent variable (y) is explained by the regressors (xs). We can calculate this as

$$R^2 = \frac{\sum_{t=1}^{T}(\hat{y}_t - \bar{y})^2}{\sum_{t=1}^{T}(y_t - \bar{y})^2} = \frac{\text{Explained (Regression) Sum of Squares}}{\text{Total Sum of Squares}}$$

Assuming that the model has an intercept:

- If the predictions are close to the actual values, we would expect R^2 to be close to 1.
- If the predictions are unrelated to the actual values, then $R^2 = 0$

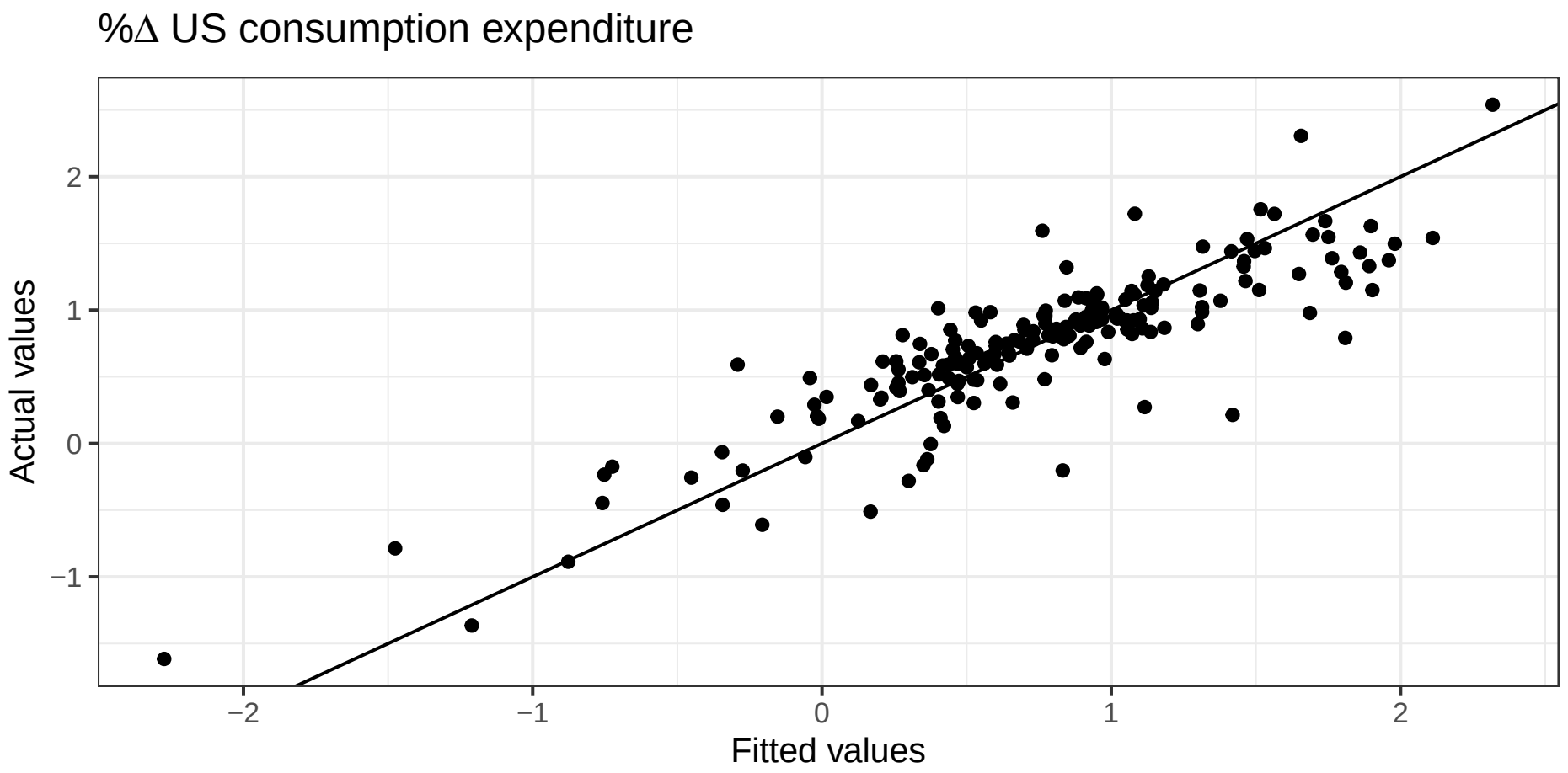
In all cases, $0 \leq R^2 \leq 1$**.**

We must be careful when comparing models on the basis of R^2

- The value of R^2 will **never decrease** when adding an extra predictor to the model and this can lead to over-fitting.
- We could be dealing with a "spurious" regression. We will get back to this at a point later.

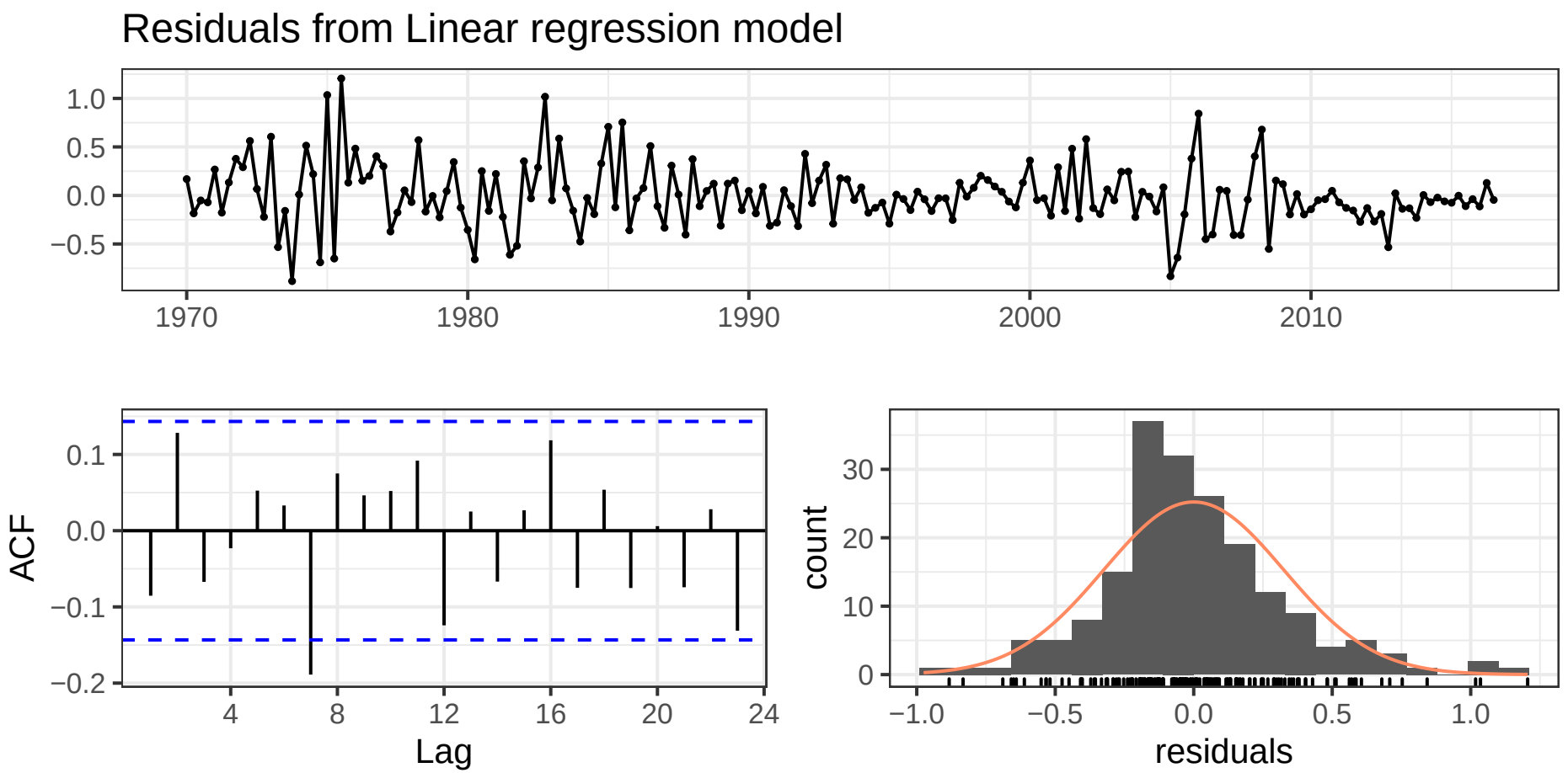
Example: US consumption expenditure

```
data.frame(Data = uschange[, "Consumption"], Fitted = fitted(fit.consMR)) %>%
    ggplot(aes(x = Data, y = Fitted)) + geom_point() + labs(x = "Fitted values",
    y = "Actual values", title = TeX("%$\\Delta$ US consumption expenditure")) +
    geom_abline(intercept = 0, slope = 1)
```



Example: US consumption expenditure

```
checkresiduals(fit.consMR, test=FALSE)
```



Section 2

Residual diagnostics

Multiple regression and forecasting

1. We assume that the model is a reasonable approximation to reality; that is, the relationship between the forecast variable and the predictor variables satisfies this linear equation.
2. We make the following assumptions about the errors $(\varepsilon_1, \ldots, \varepsilon_T)$:
   - ε_t are uncorrelated and zero mean otherwise the forecasts will be systematically biased.
   - ε_t are uncorrelated with each $x_{j,t}$ otherwise the forecasts will be inefficient, as there is more information in the data that can be exploited.
   - They are unrelated to the predictor variables otherwise there would be more information that should be included in the systematic part of the model.

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

Residual plots

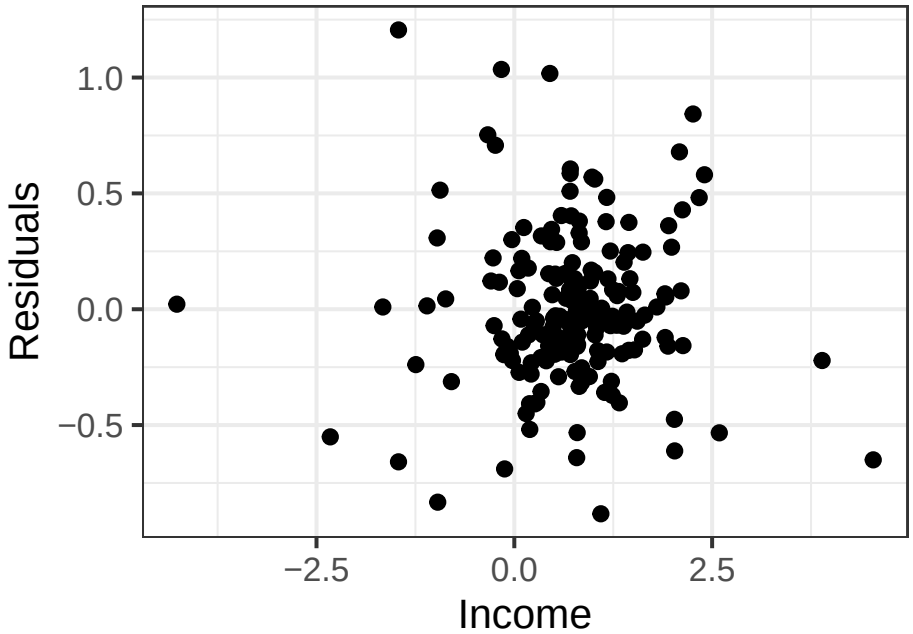
Useful for spotting outliers and whether the linear model was appropriate.

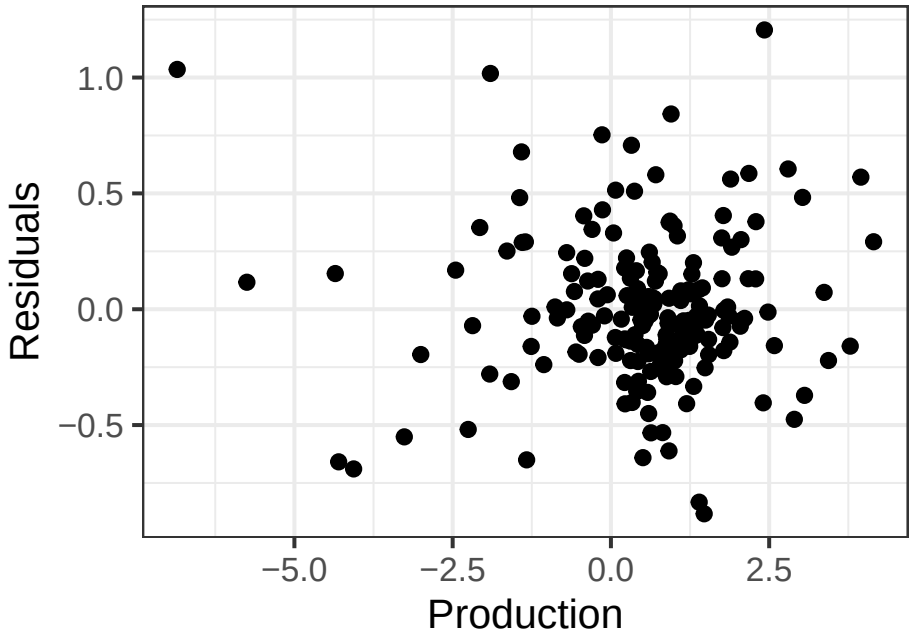
- Scatterplot of residuals ε_t against each predictor $x_{j,t}$.
- Scatterplot residuals against the fitted values $\hat{y}_t$
- Expect to see scatterplots resembling a horizontal band with no values too far from the band and no patterns such as curvature or increasing spread.

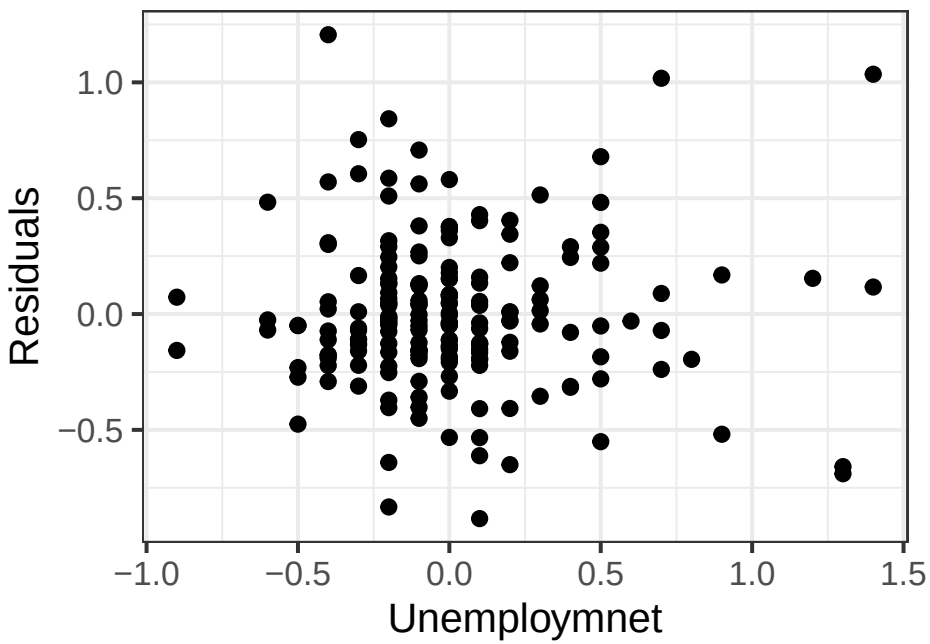
Residual patterns

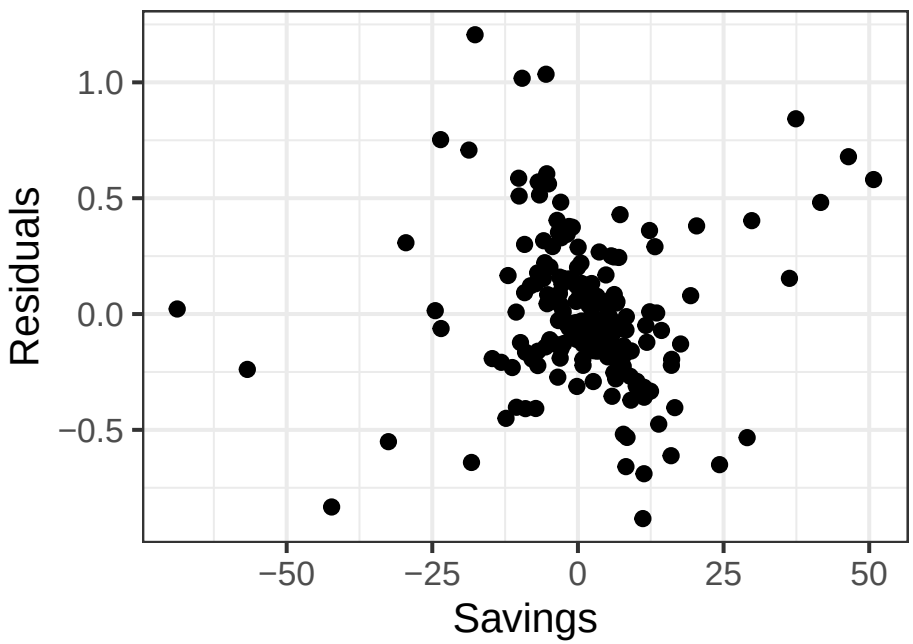
- If a plot of the residuals vs any predictor **in** the model shows a pattern, then the relationship is nonlinear.
- If a plot of the residuals vs any predictor **not** in the model shows a pattern, then the predictor should be added to the model.
- If a plot of the residuals vs fitted values shows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

Residual patterns









Breusch-Godfrey test

H_0 : There is no autocorrelation up to lag p.

OLS regression:

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

Auxiliary regression:

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

If R^2 statistic is calculated for the auxiliary model, then

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

Here, we are testing that there is no serial correlation up to lag p.
T = length of series.

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

US consumption again

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

```
##
##  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.

Section 3

(Some) Useful predictors for linear models

Linear trend

Given the general form:

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

We can introduce a trend to the model by including $x_t = t$ as a regressor,

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

where $t = 1, 2, \ldots, T$

A trend variable can be specified in the `tslm()` function using the `trend` predictor.

Why would you want to include a trend?

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**.

Example

Suppose we have quarterly retail sales data and suspect that there might be seasonality in our data (e.g. Q4 might have unusually high sales figures since we have Thanksgiving, Black Friday, Cyber Monday, and Christmas in Nov. & Dec.)

	$Q_{1,t}$	$Q_{2,t}$	$Q_{3,t}$
2000 Q1	1	0	0
2000 Q2	0	1	0
2000 Q3	0	0	1
2000 Q4	0	0	0
2001 Q1	1	0	0
2001 Q2	0	1	0
2001 Q3	0	0	1
2001 Q4	0	0	0
⋮	⋮	⋮	⋮

Dummy variables

Beware of the dummy variable trap!

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).

- 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.
- If we omit one category, the **coefficients of the remaining dummies are relative to that 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

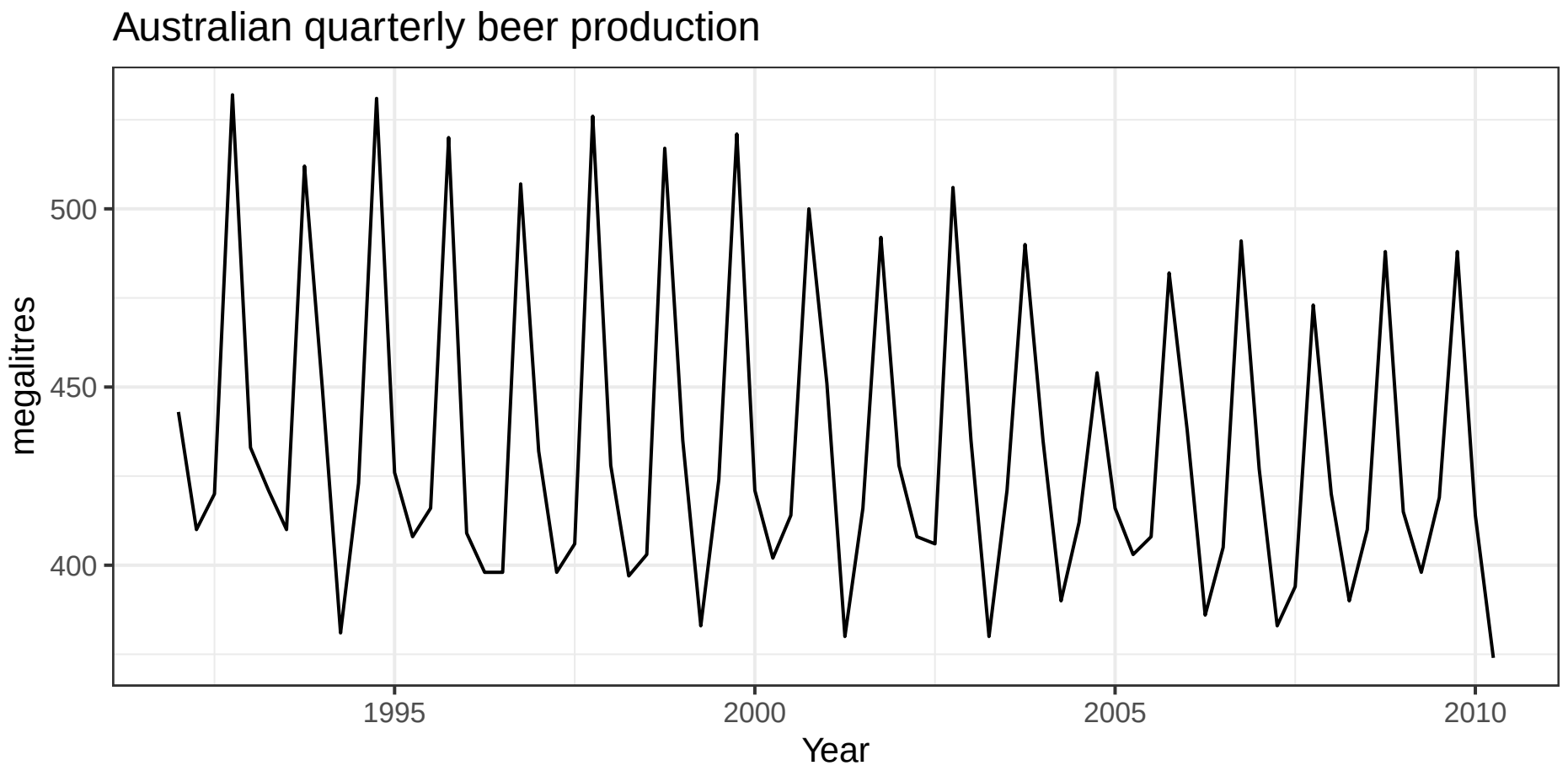
- If there is an outlier, you can use a dummy variable (taking value 1 for that observation and 0 elsewhere) to remove its effect.

Public holidays

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

Beer production revisited

```
beer2 <- window(ausbeer, start=1992)
autoplot(beer2) + labs(x = "Year", y = "megalitres",
                       title = "Australian quarterly beer production")
```

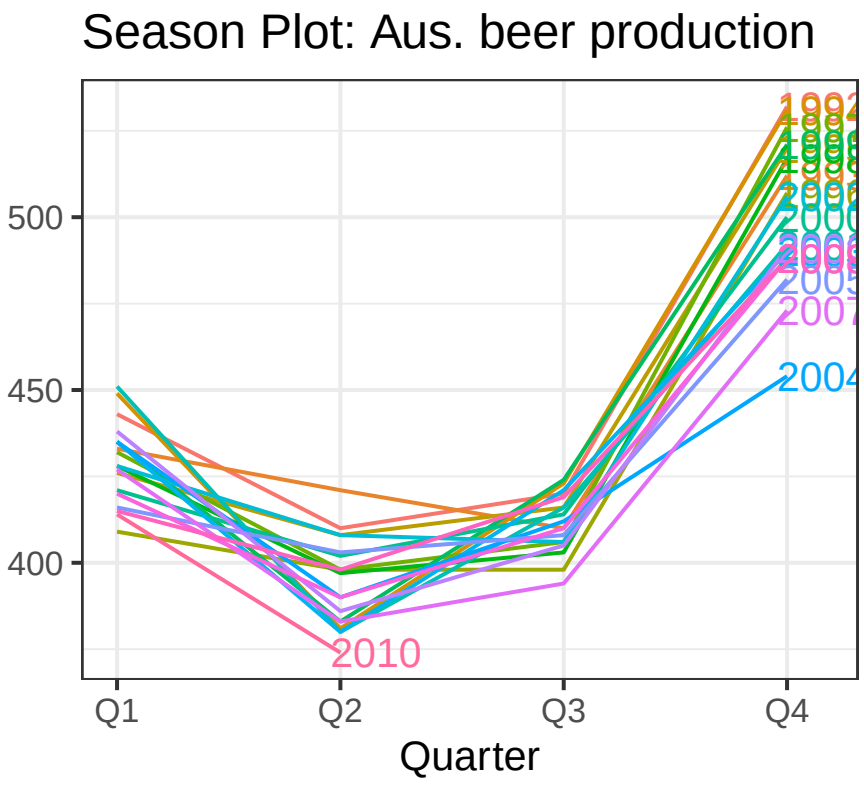


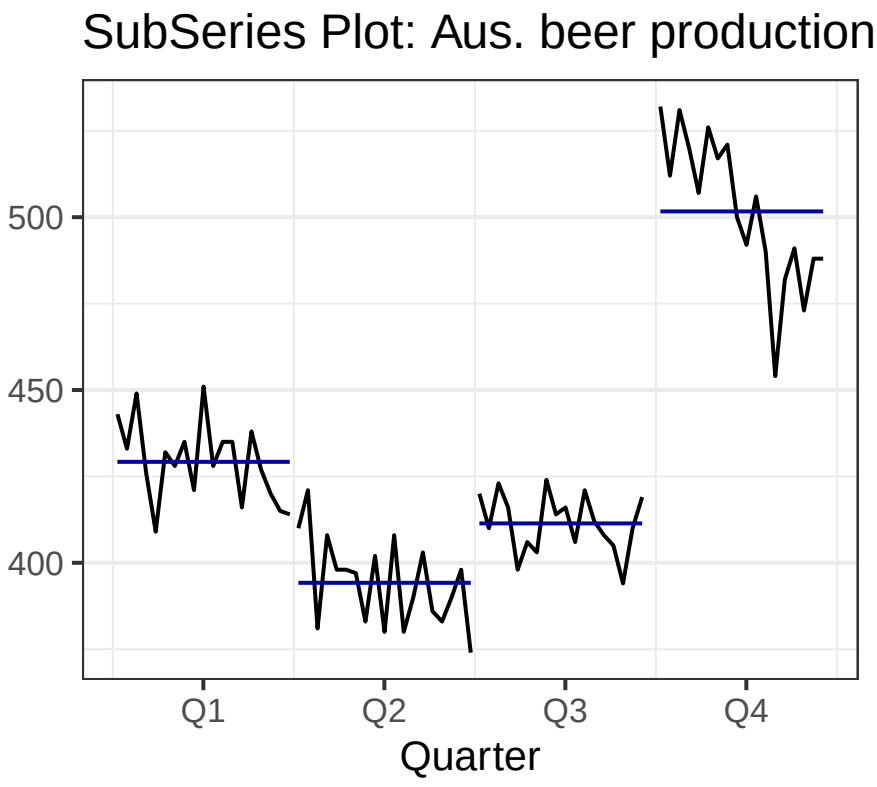
Beer production revisited

Regression model

$$y_t = \beta_0 + \beta_1 t + \gamma_1 Q_{2,t} + \gamma_2 Q_{3,t} + \gamma_3 Q_{4,t} + \varepsilon_t$$

- $Q_{i,t} = 1$ if t is in quarter i and 0 otherwise , for $i \in [2, 4]$.





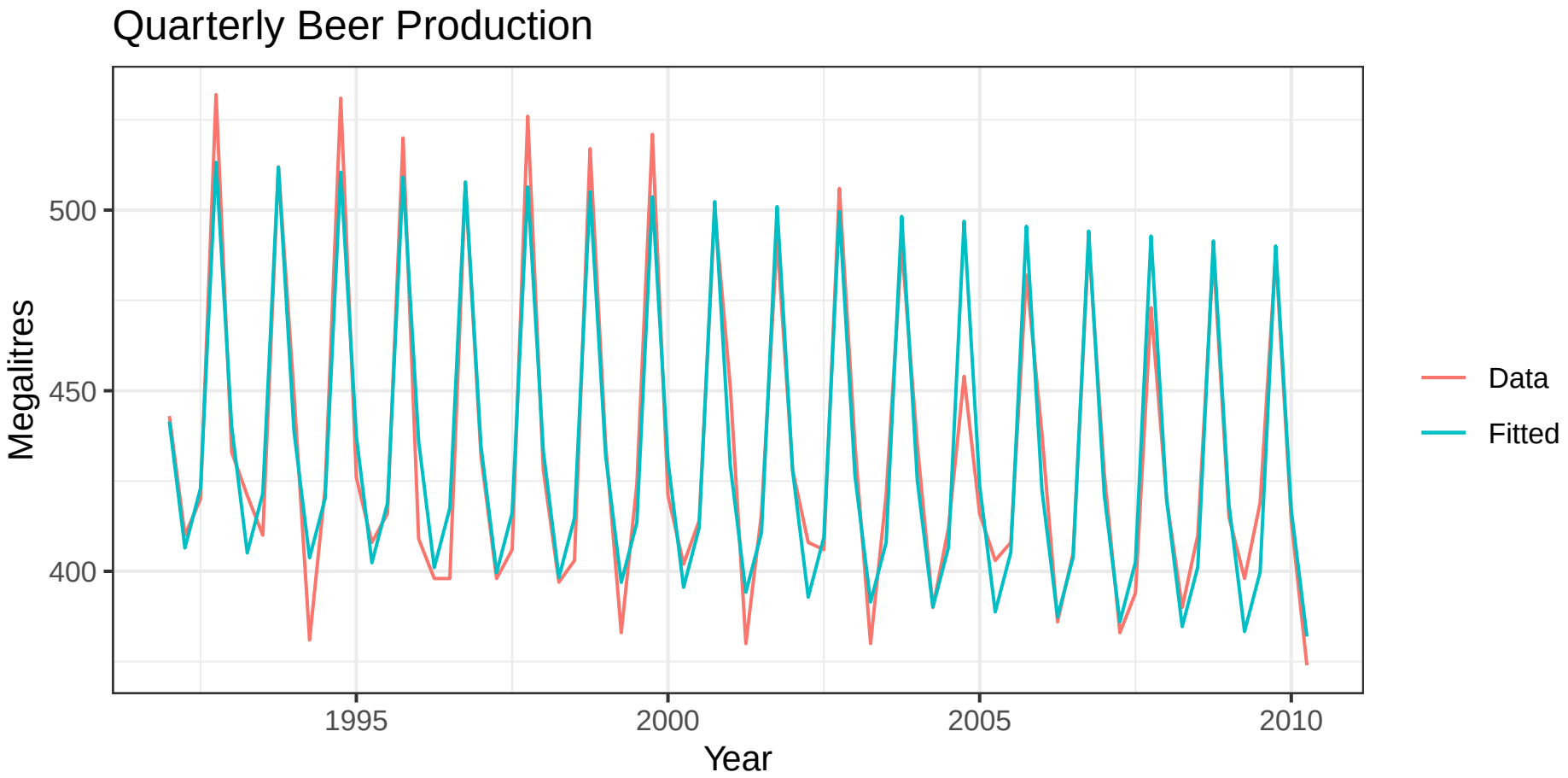
Beer production revisited

```
# Model with trend and seasonal dummies
fit.beer <- tslm(beer2 ~ trend + season)
summary(fit.beer)
```

```
##
## Call:
## tslm(formula = beer2 ~ 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
```

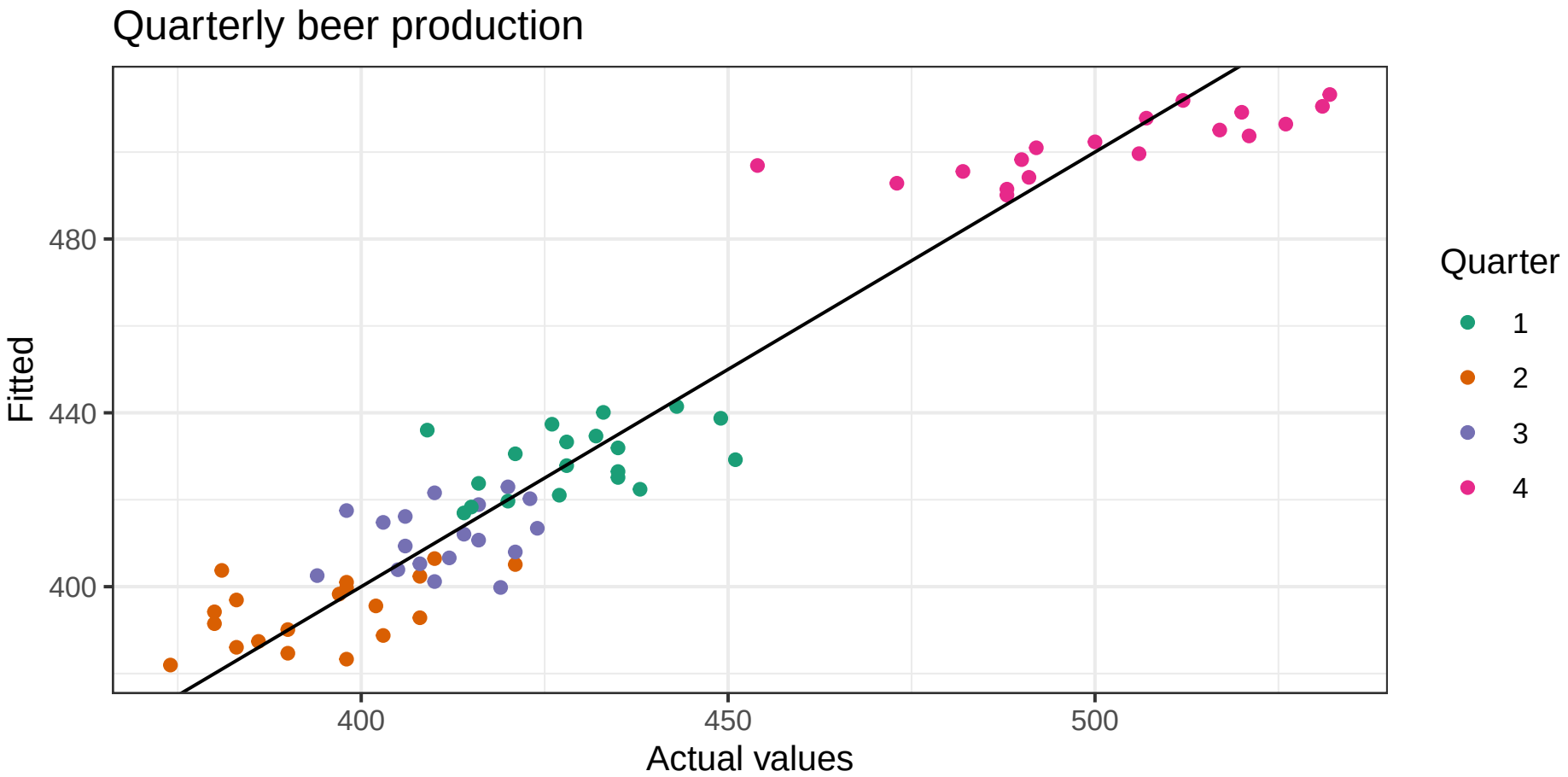
Beer production revisited

```
autoplot(beer2, series="Data") +
  autolayer(fitted(fit.beer), series="Fitted") +
 labs(x = "Year", y = "Megalitres", title = "Quarterly Beer Production") +
  guides(colour=guide_legend(title=" "))
```



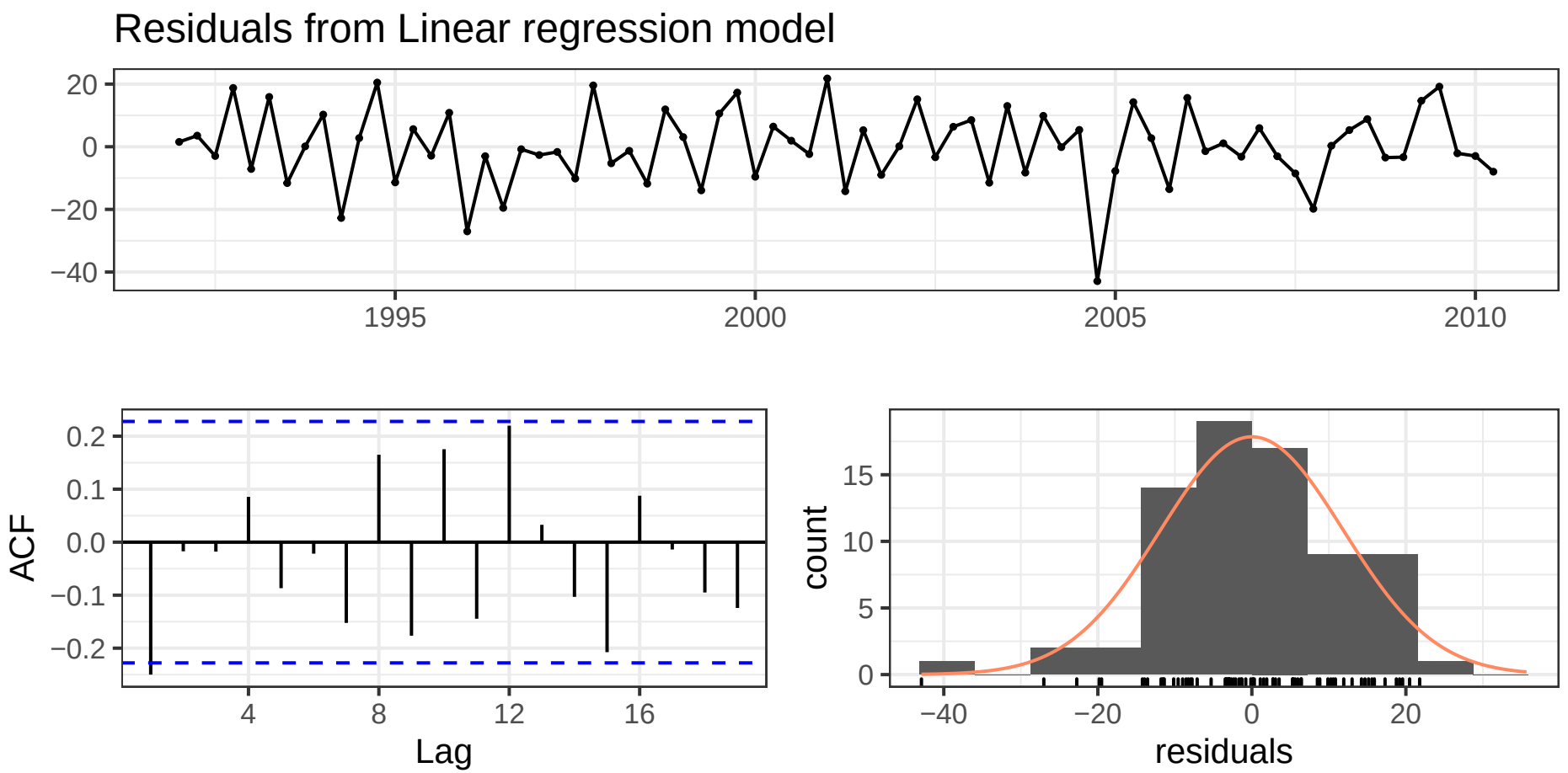
Beer production revisited

```
data.frame(Data=beer2, Fitted=fitted(fit.beer)) %>%
  ggplot(aes(x=Data, y=Fitted, colour=as.factor(cycle(beer2)))) +
  geom_point() + labs(y = "Fitted", x = "Actual values",
                      title = "Quarterly beer production") +
  scale_colour_brewer(palette="Dark2", name="Quarter") +
  geom_abline(intercept=0, slope=1)
```



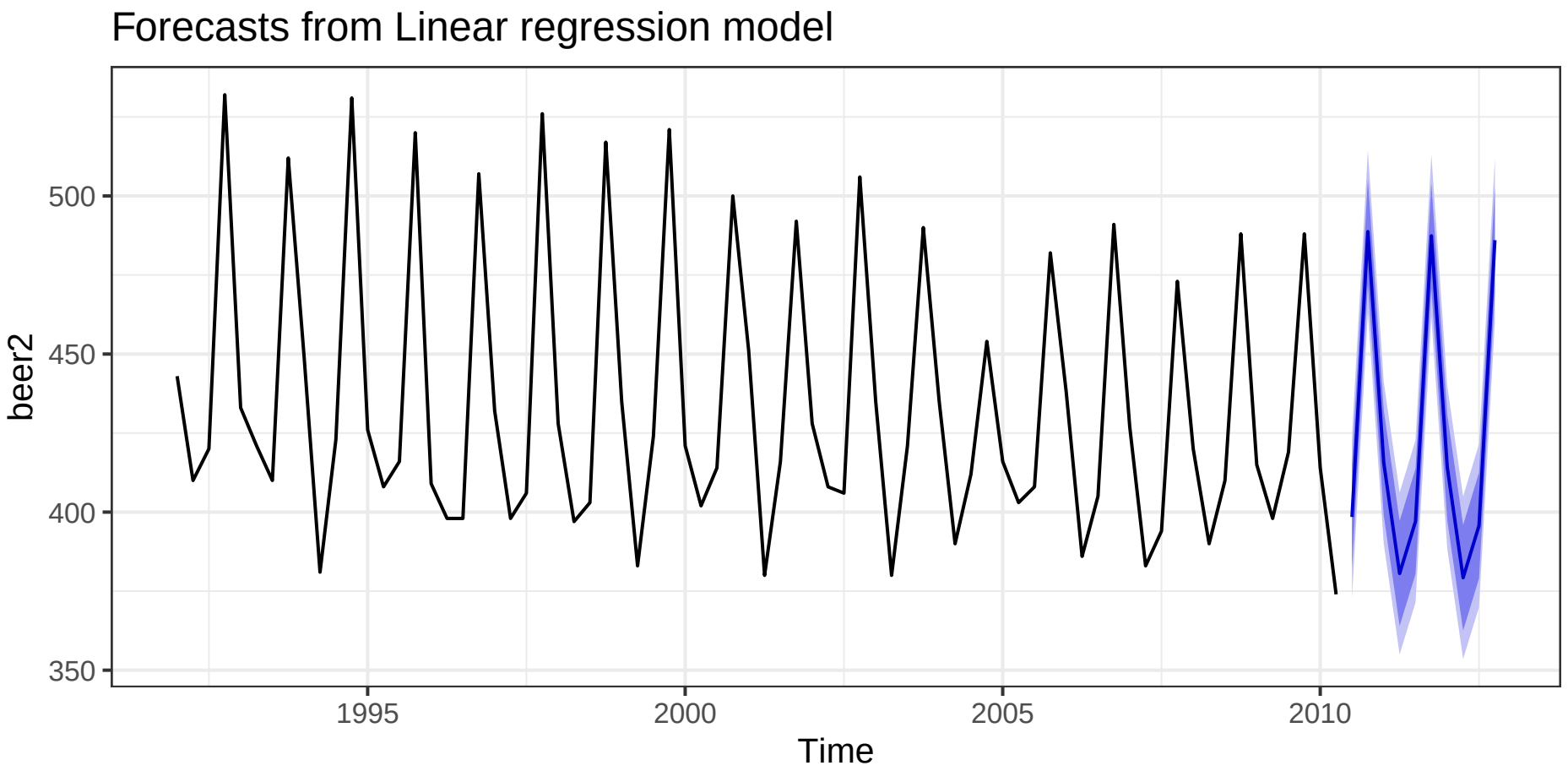
Beer production revisited

```
checkresiduals(fit.beer, test=FALSE)
```



Forecasting Beer production

```
fit.beer %>% forecast %>% autoplot
```



Fourier series

Periodic seasonality can be handled using pairs of Fourier terms:

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

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

- where `m` is the seasonal period.
- 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

```
#maximum allowed is K = m/2
fourier.beer <- tslm(beer2 ~ trend + fourier(beer2, K=2))
summary(fourier.beer)
```

```
##
## Call:
## tslm(formula = beer2 ~ trend + fourier(beer2, 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(beer2, K = 2)S1-4    8.91082    2.01125   4.430 3.45e-05 ***
## fourier(beer2, K = 2)C1-4   53.72807    2.01125  26.714  < 2e-16 ***
## fourier(beer2, 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

- the effect of an event lasts for only 1 period. We use a dummy variable to capture this.
- equivalent to a dummy variable for handling an outlier.

Steps

- the intervention has an immediate and permanent effect. The intervention causes a level shift.
- dummy variable takes value 0 before the intervention and 1 afterwards.

Change of slope

- the intervention causes a permanent effect and changes the slope.
- we use a piecewise trend here. The trend is no longer linear.
- variables take values 0 before the intervention and values $\{1, 2, 3, \dots\}$ afterwards.
- Interact the dummy with your regressors?

Distributed lags

Lagged values of a predictor.

Example: x is advertising which has a delayed effect

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

Section 4

Selecting predictors and forecast evaluation

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. 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.
- using R^2 to determine whether a model will give good predictions will lead to overfitting.

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

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 $\hat{\sigma}_e$.

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

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.

Alternatively,

$$\text{AIC} = T\log\left(\frac{\text{SSE}}{T}\right) + 2(k+2),$$

- 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 (when $T \to \infty$) 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.

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

As with the AIC, the AIC_C should be **minimized**.

Schwarz's 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.

Alternatively,

$$\text{BIC} = T\log\left(\frac{\text{SSE}}{T}\right) + (k+2)\log(T)$$

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

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 potential regressors.
- Try dropping 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.
- If the number of potential predictors is too large, then the backwards stepwise regression will not work and *forward stepwise* regression can be used instead.

Choosing regression variables

Forward stepwise regression

- This procedure starts with a model that includes only the intercept.
- Predictors are added one at a time, and the one that most improves the measure of predictive accuracy is retained in the model.
- The procedure is repeated until no further improvement can be achieved.

Cross-validation

```
model <- matrix(NA,nrow = 5, ncol = 5,
                dimnames = list(paste0("Model ", 1:5),
                                c("CV","AIC","AICc", "BIC", "AdjR2")))
model[1,] <- tslm(Consumption ~ Income + Production + Unemployment +
                    Savings, data=uschange) %>% CV()
model[2,] <- tslm(Consumption ~ Income + Production + Unemployment,
                  data=uschange) %>% CV()
model[3,] <- tslm(Consumption ~ Income + Production + Savings,
                  data=uschange) %>% CV()
model[4,] <- tslm(Consumption ~ Income + Unemployment + Savings,
                  data=uschange) %>% CV()
model[5,] <- tslm(Consumption ~ Production + Unemployment + Savings,
                  data=uschange) %>% CV()

knitr::kable(model,digits = 3)
```

	CV	AIC	AICc	BIC	AdjR2
Model 1	0.116	-409.298	-408.831	-389.911	0.749
Model 2	0.278	-243.164	-242.832	-227.008	0.386
Model 3	0.118	-407.467	-407.135	-391.311	0.745
Model 4	0.116	-408.094	-407.763	-391.939	0.746
Model 5	0.293	-234.373	-234.042	-218.218	0.356

Section 5

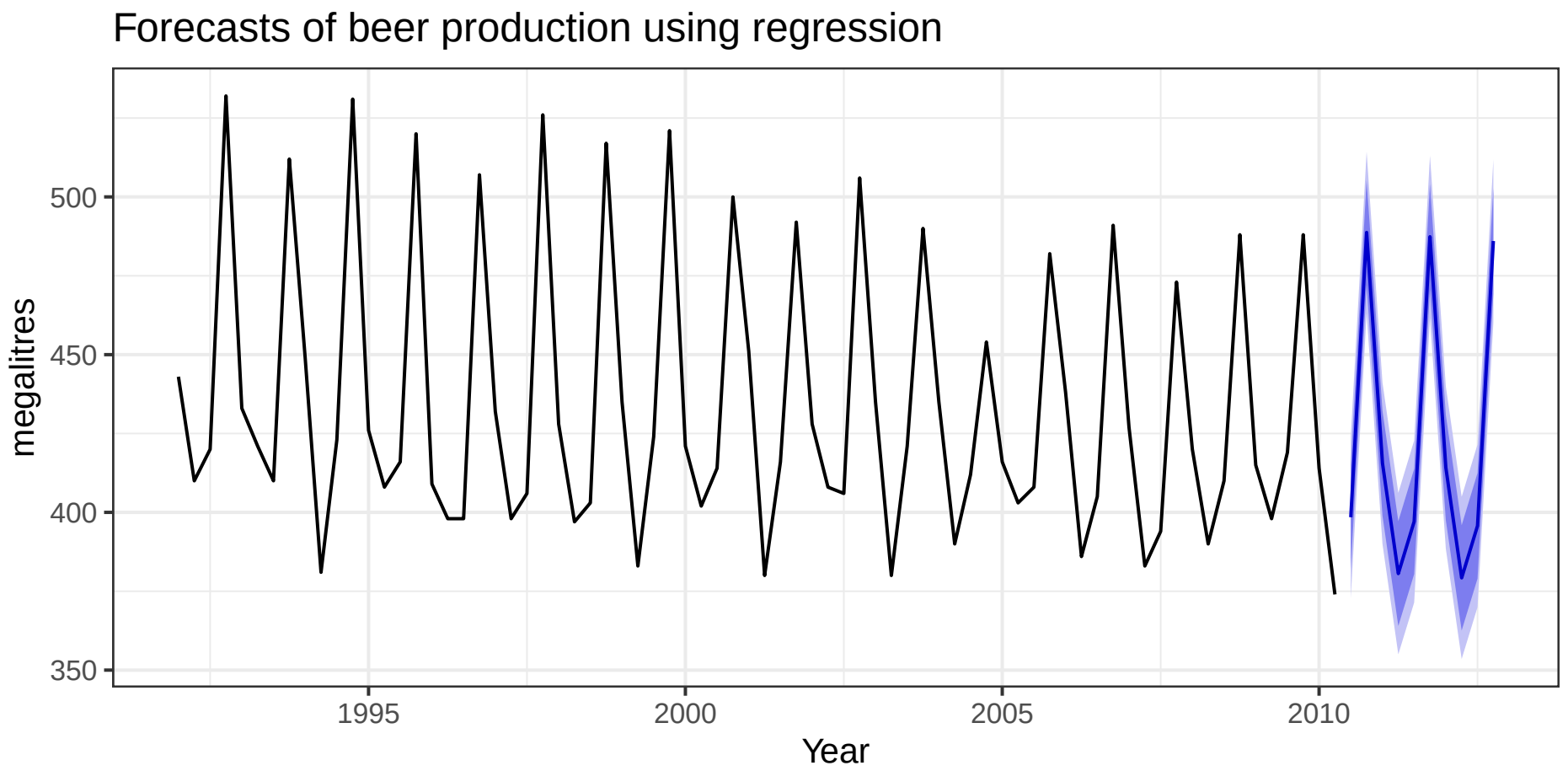
Forecasting with regression

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 forecasted.

Beer production

```
fit.beer <- tslm(beer2 ~ trend + season)
fcast <- forecast(fit.beer)
autoplot(fcast) + labs(x = "Year", y = "megalitres",
      title = "Forecasts of beer production using regression")
```



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.

US Consumption

Example

A US policy maker is interested in comparing the predicted change in consumption when there is a *constant growth of 1% and 0.5% respectively for income and savings with no change in the employment rate*, versus a *respective decline of 1% and 0.5%*, for **each of the four quarters following the end of the sample**.

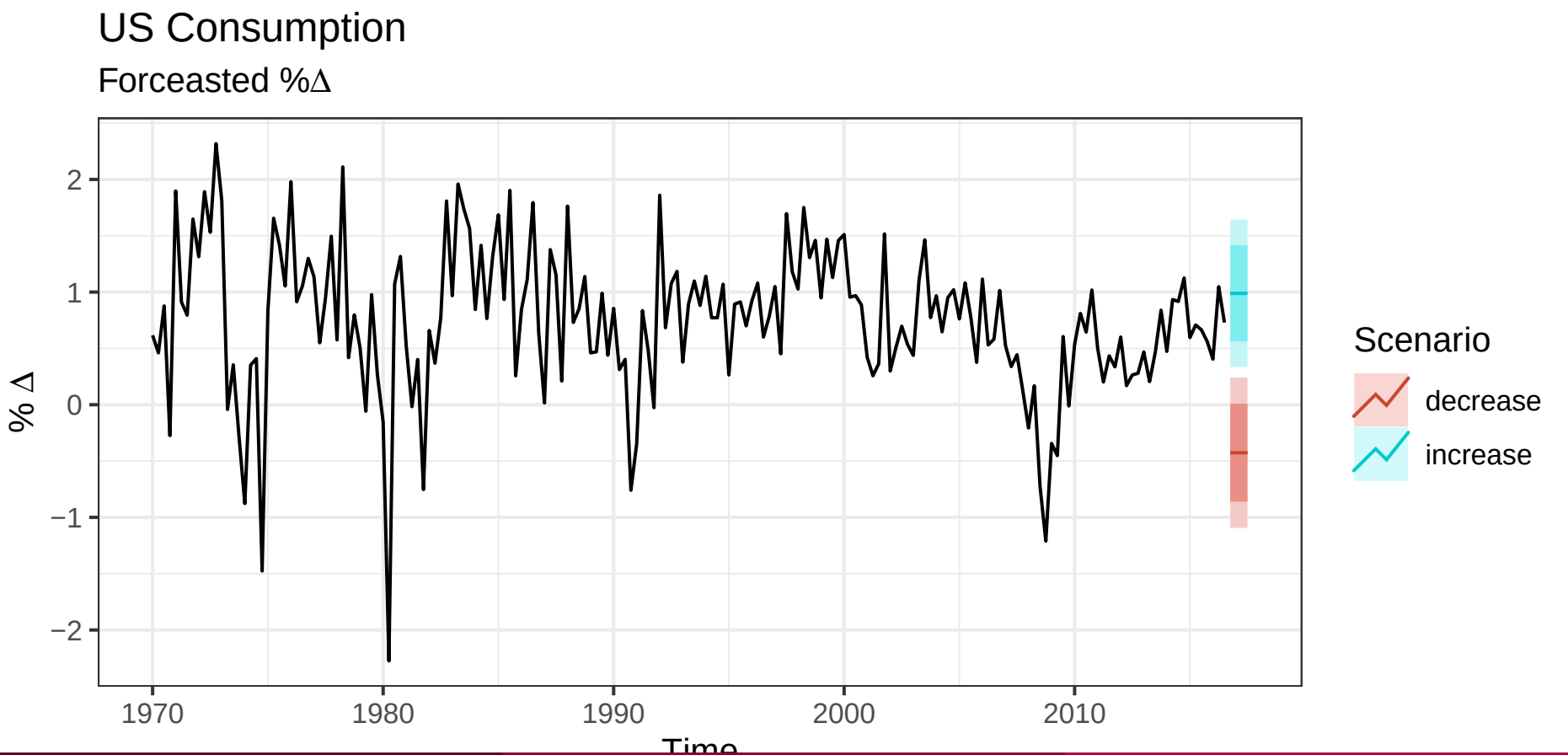
```
fit.consBest <- tslm(Consumption ~ Income + Savings + Unemployment,
                     data = uschange)
h <- 4
newdata <- data.frame(Income = rep(1, h),
    Savings = rep(0.5, h), Unemployment = rep(0,h))
fcast.up <- forecast(fit.consBest, newdata = newdata)
newdata <- data.frame(Income = rep(-1,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

Note

- The prediction intervals for scenario based forecasts do not include the uncertainty associated with the future values of the predictor variables.
- They assume that the values of the predictors are known in advance.

US Consumption

```
autoplot(uschange[,1]) +
  labs(y = TeX('% $\\Delta$'), title = "US Consumption",
       subtitle = TeX('Forceasted %$\\Delta$') ) +
  autolayer(fcast.up, PI = TRUE, series = "increase") +
  autolayer(fcast.down, PI = TRUE, series = "decrease") +
  guides(colour = guide_legend(title = "Scenario"))
```



Building a predictive regression model

- The great advantage of regression models is that they can be used to capture important relationships between the forecast variable of interest and the predictor variables.
- A major challenge however, is that in order to generate ex-ante forecasts, the model requires future values of each predictor.
- If scenario based forecasting is of interest then these models are extremely useful.
- If *ex-ante* forecasting of y is the main focus, then you will need future/forecasted values of the xs, which might be challenging.
- If getting forecasts of predictors is difficult, you can use lagged predictors instead.

Building a predictive regression model

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

Scrolling the model forward

$$y_{t+h} = \beta_0 + \beta_1 x_{1,t} + \cdots + \beta_k x_{k,t} + \varepsilon_{t+h}$$

for $h = 1, 2, \ldots$

- Notice that the predictor set, $\{x\}$, is formed by values of the$x$s observed h time periods prior to observing y.
- Therefore when the estimated model is projected into the future, i.e., beyond the end of the sample, T, all predictor values are available.

Section 6

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 ± 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.

Thinking back to the Dummy variable trap

Suppose you have quarterly data and use four dummy variables, d_1, d_2, d_3, d_4. Then $d_4 = 1 - d_1 - d_2 - d_3$ so there is perfect correlation between d_4 and $d_1 + d_2 + d_3$. Knowing d_1, d_2, d_3 will help us to pefectly predict d_4

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.
- the uncertainty associated with individual regression coefficients will be large. That is the variance is inflated.
- 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.

Section 7

Nonlinear Regressions

Log-Log Model

$$\log y = \beta_0 + \beta_1 \log x + \varepsilon$$

While this provides a non-linear functional form, the model is still linear in the parameters.

In this model, the slope, β_1 can be interpreted as an elasticity. In fact, β_1 is the anticipated percentage change in y resulting from a 1% increase in the x variable.

How?

$$\frac{d \log y}{dx} = \beta_1 \frac{\log(x)}{dx}$$
$$\Rightarrow \frac{1}{y} \cdot \frac{dy}{dx} = \beta_1 \cdot \frac{1}{x}$$
$$\Rightarrow \beta_1 = \frac{x}{y} \cdot \frac{dy}{dx}$$

Other log tranformations

Log-linear form is specified by only transforming the forecast variable.

$$\log y = \beta_0 + \beta_1 x + \varepsilon$$

Linear-log form is obtained by transforming the predictor.

$$y = \beta_0 + \beta_1 \log x + \varepsilon$$

Working with zeros

- Recall that in order to perform a logarithmic transformation to a variable, all of its observed values must be greater than zero.
- In the event that variable x contains zeros, we use the transformation

$$\log(x + 1)$$

Nonlinear trend

Piecewise linear trend with bend at τ

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

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

Quadratic or higher order trend

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

NOT RECOMMENDED!: to use quadratic or higher order trends in forecasting. When they are extrapolated, the resulting forecasts are often unrealistic.