Chapter 7 Time Series Regression Models

AG

19/01/2021

The Linear Model

Linear Regression

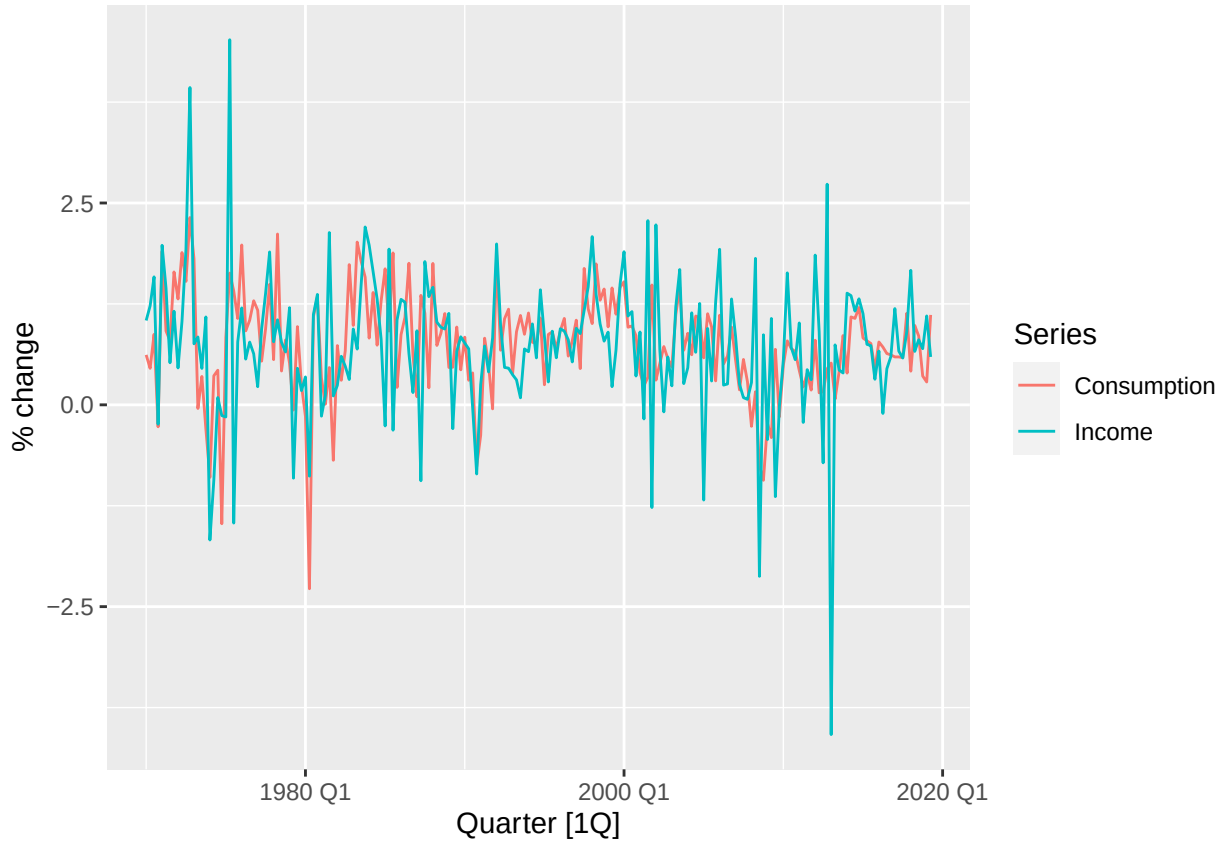
```
us_change
```

```
## # A tsibble: 198 x 6 [1Q]
##    Quarter Consumption Income Production Savings Unemployment
##      <qtr>       <dbl>  <dbl>      <dbl>   <dbl>        <dbl>
##  1 1970 Q1       0.619  1.04      -2.45    5.30         0.9
##  2 1970 Q2       0.452  1.23      -0.551   7.79         0.5
##  3 1970 Q3       0.873  1.59      -0.359   7.40         0.5
##  4 1970 Q4      -0.272 -0.240     -2.19    1.17         0.700
##  5 1971 Q1       1.90   1.98       1.91    3.54        -0.100
##  6 1971 Q2       0.915  1.45       0.902   5.87        -0.100
##  7 1971 Q3       0.794  0.521      0.308  -0.406        0.100
##  8 1971 Q4       1.65   1.16       2.29   -1.49         0
##  9 1972 Q1       1.31   0.457      4.15   -4.29        -0.2
## 10 1972 Q2       1.89   1.03       1.89   -4.69        -0.100
## # ... with 188 more rows
```

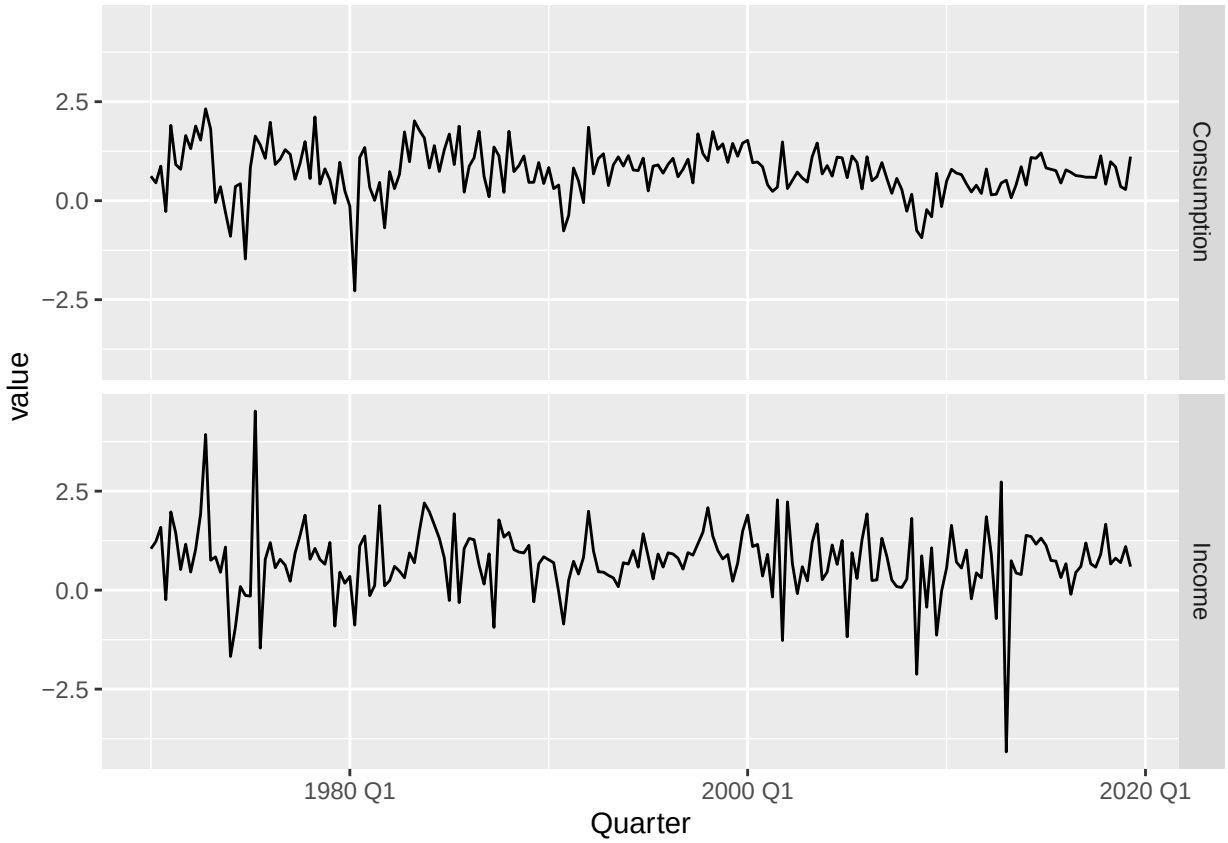
```
us_change %>%
  pivot_longer(c(Consumption, Income), names_to = "Series")
```

```
## # A tsibble: 396 x 6 [1Q]
## # Key:       Series [2]
##    Quarter Production Savings Unemployment Series       value
##      <qtr>      <dbl>   <dbl>        <dbl> <chr>        <dbl>
##  1 1970 Q1     -2.45     5.30         0.9   Consumption  0.619
##  2 1970 Q1     -2.45     5.30         0.9   Income       1.04
##  3 1970 Q2     -0.551    7.79         0.5   Consumption  0.452
##  4 1970 Q2     -0.551    7.79         0.5   Income       1.23
##  5 1970 Q3     -0.359    7.40         0.5   Consumption  0.873
##  6 1970 Q3     -0.359    7.40         0.5   Income       1.59
##  7 1970 Q4     -2.19     1.17         0.700 Consumption -0.272
##  8 1970 Q4     -2.19     1.17         0.700 Income      -0.240
##  9 1971 Q1      1.91     3.54        -0.100 Consumption  1.90
## 10 1971 Q1      1.91     3.54        -0.100 Income       1.98
## # ... with 386 more rows
```

```
us_change %>%
  pivot_longer(c(Consumption, Income), names_to = "Series") %>%
  autoplot(value) +
  labs(y = "% change")
```

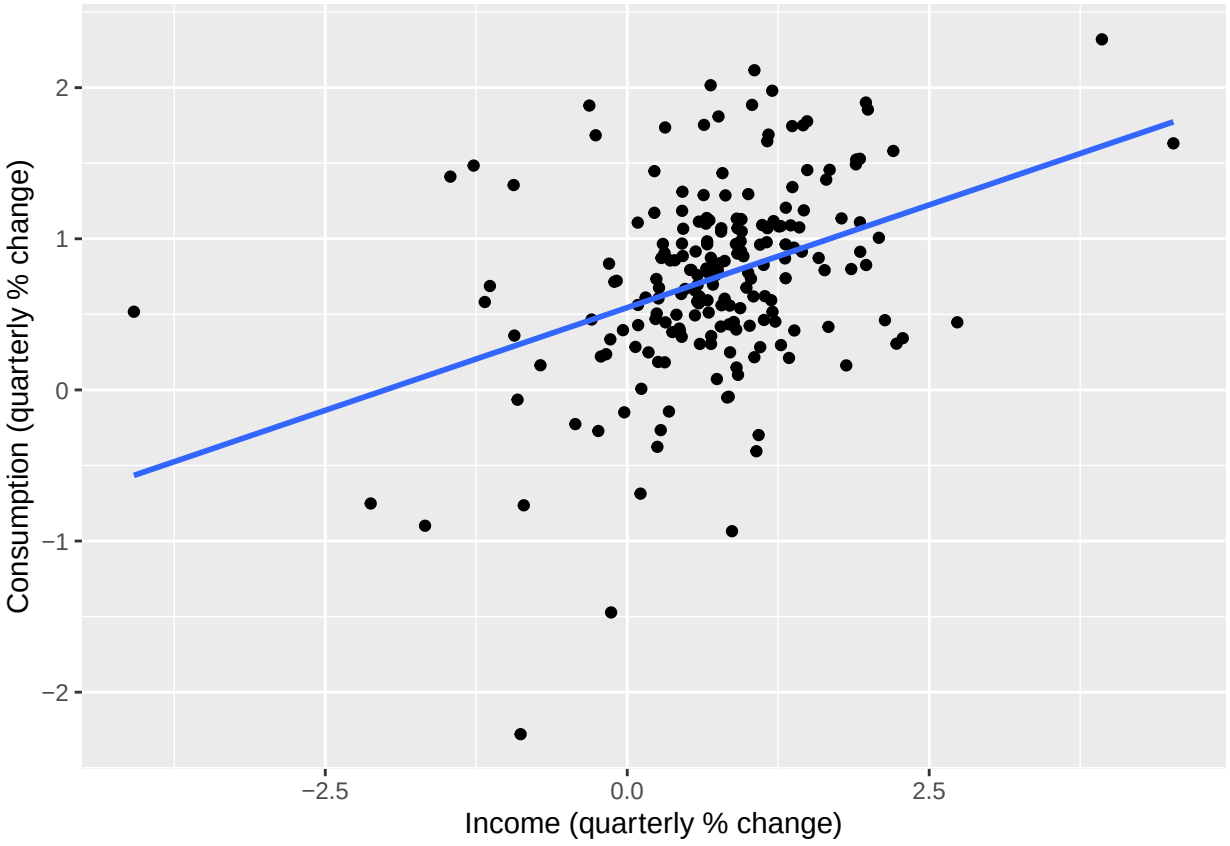



```
us_change %>% pivot_longer(c(Consumption, Income), names_to = "Series") %>% ggplot(aes(x = Quarter, y =
```




```
us_change %>%
  ggplot(aes(x = Income, y = Consumption)) +
  labs(y = "Consumption (quarterly % change)", x = "Income (quarterly % change)") +
  geom_point() +
  geom_smooth(method = "lm", se = FALSE)
```

```
## `geom_smooth()` using formula 'y ~ x'
```

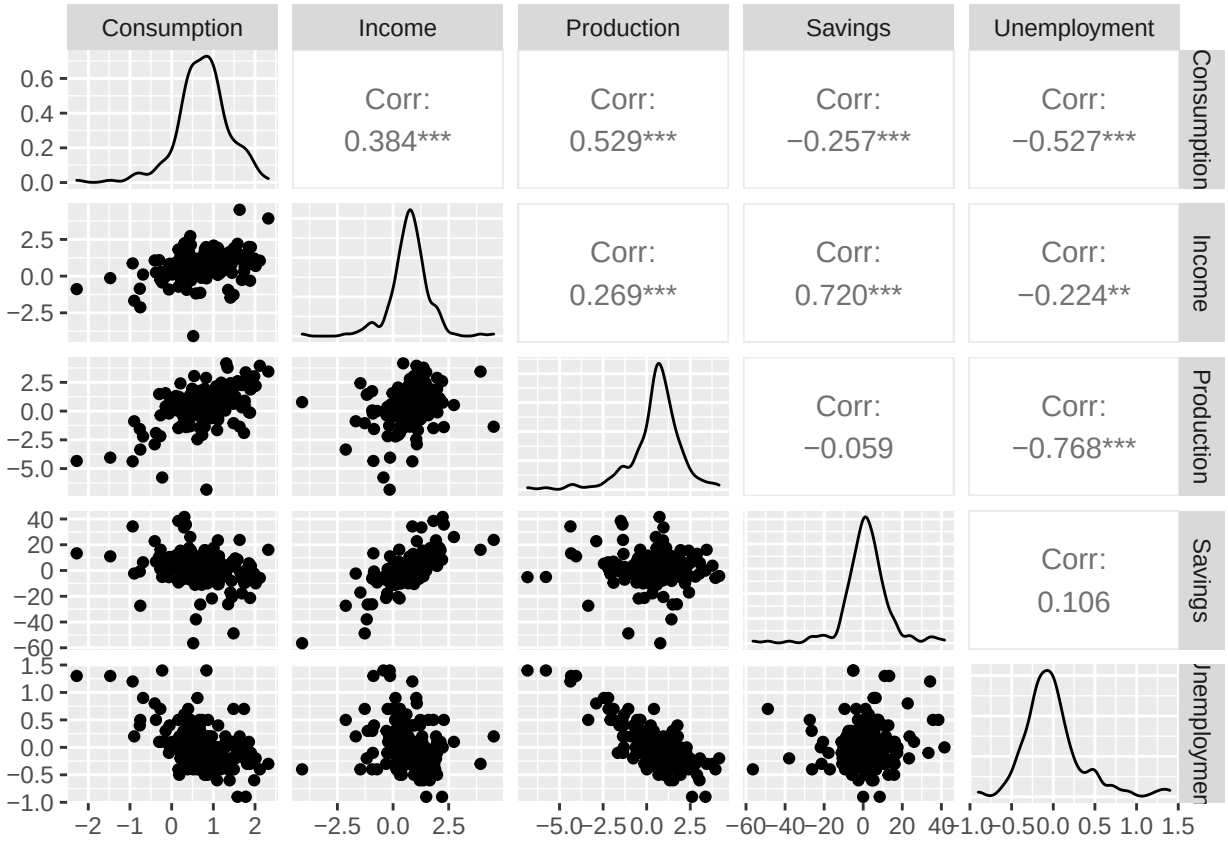



```
us_change %>%
  model(TSLM(Consumption ~ Income)) %>%
  report()
```

```
## Series: Consumption
## Model: TSLM
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.58236 -0.27777  0.01862  0.32330  1.42229
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.54454    0.05403  10.079  < 2e-16 ***
## Income       0.27183    0.04673   5.817  2.4e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.5905 on 196 degrees of freedom
## Multiple R-squared: 0.1472,  Adjusted R-squared: 0.1429
## F-statistic: 33.84 on 1 and 196 DF, p-value: 2.4022e-08
```

```
us_change %>%
  GGally::ggpairs(columns = 2:6)
```

```
## Registered S3 method overwritten by 'GGally':
##   method from
##   +.gg   ggplot2
```



Least Square Estimation

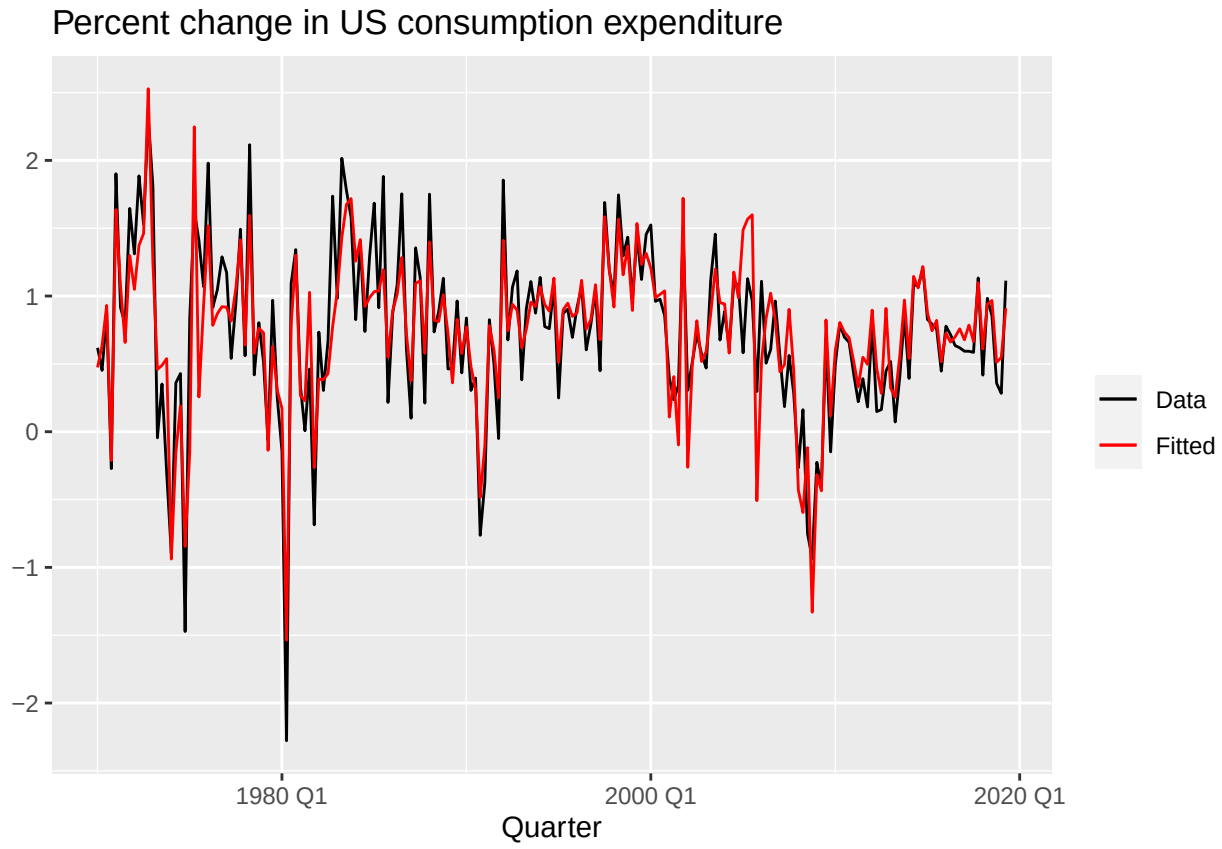
```
fit.consMR <- us_change %>%
  model(tslm = TSLM(Consumption ~ Income + Production + Unemployment + Savings))
report(fit.consMR)
```

```
## Series: Consumption
## Model: TSLM
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.90555 -0.15821 -0.03608  0.13618  1.15471
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)   0.253105   0.034470   7.343 5.71e-12 ***
## Income        0.740583   0.040115  18.461  < 2e-16 ***
## Production    0.047173   0.023142   2.038   0.0429 *
## Unemployment -0.174685   0.095511  -1.829   0.0689 .
```

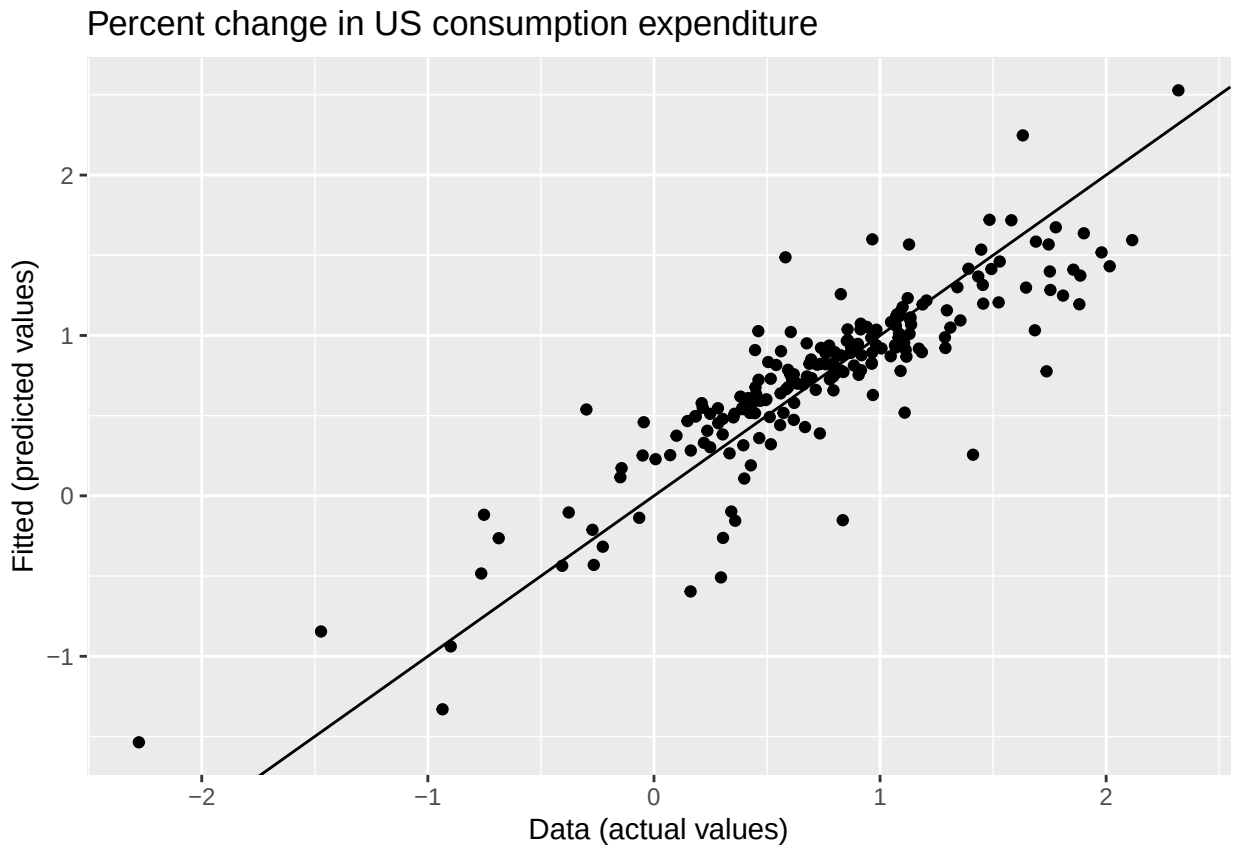
```
## Savings      -0.052890   0.002924 -18.088  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3102 on 193 degrees of freedom
## Multiple R-squared: 0.7683,  Adjusted R-squared: 0.7635
## F-statistic:   160 on 4 and 193 DF, p-value: < 2.22e-16
```

###Fitted Values

```
augment(fit.consMR) %>%
  ggplot(aes(x = Quarter)) +
  geom_line(aes(y = Consumption, colour = "Data")) +
  geom_line(aes(y = .fitted, colour = "Fitted")) +
  labs(y = NULL, title = "Percent change in US consumption expenditure") +
  scale_color_manual(values=c(Data="black",Fitted="red")) +
  guides(colour = guide_legend(title = NULL))
```



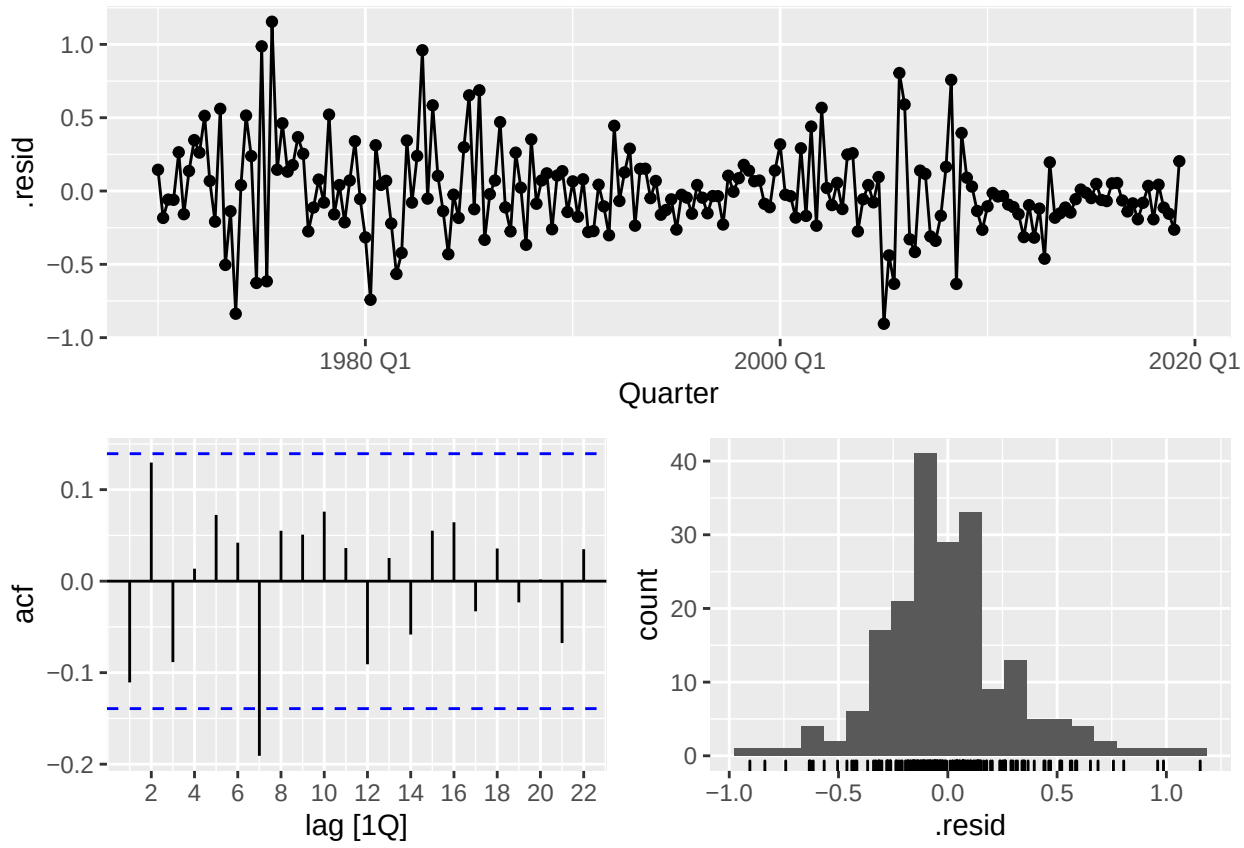

```
augment(fit.consMR) %>%
  ggplot(aes(x = Consumption, y = .fitted)) +
  geom_point() +
  labs(y = "Fitted (predicted values)",
       x = "Data (actual values)",
       title = "Percent change in US consumption expenditure") +
  geom_abline(intercept = 0, slope = 1)
```



Evaluating the regression model

Plotting Residuals

```
fit.consMR %>% gg_tsresiduals()
```

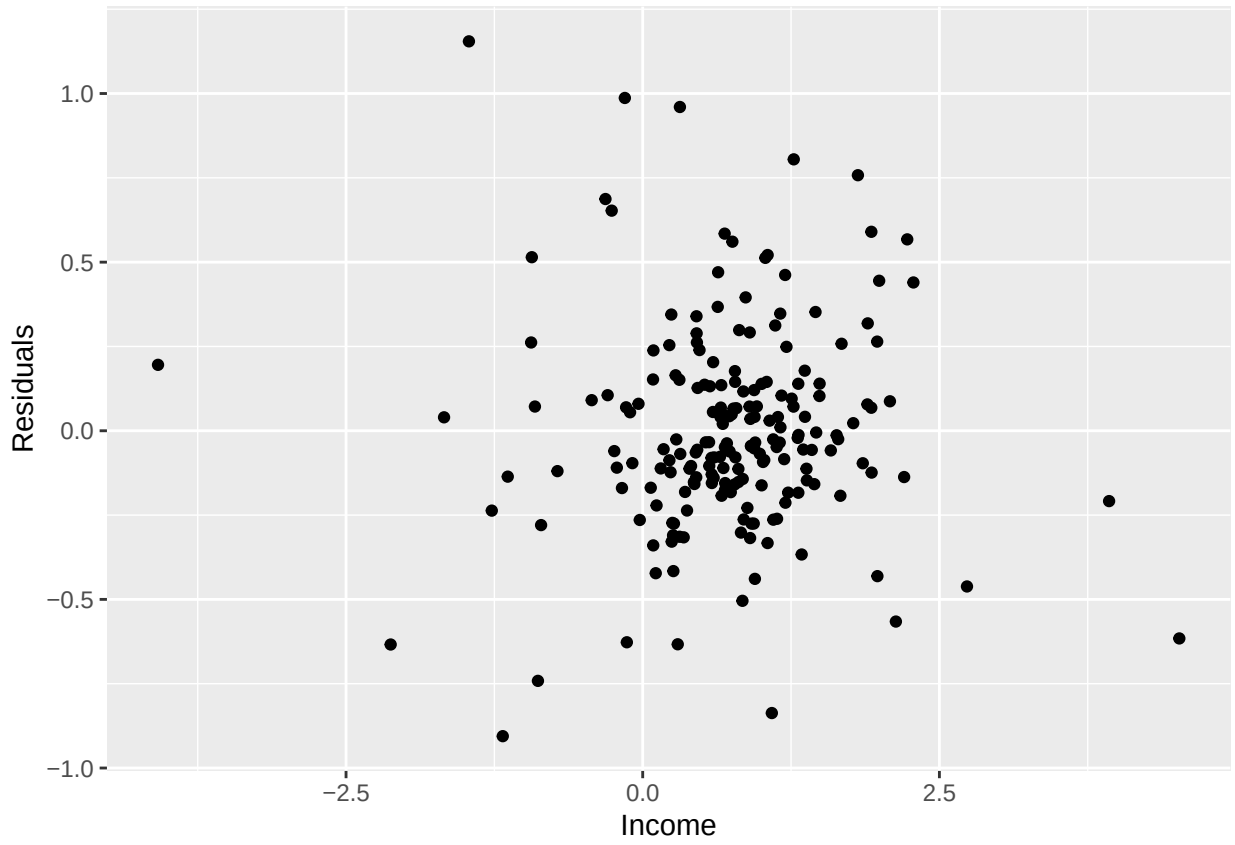



```
augment(fit.consMR) %>% features(.innov, ljung_box, lag = 10, dof = 5)
```

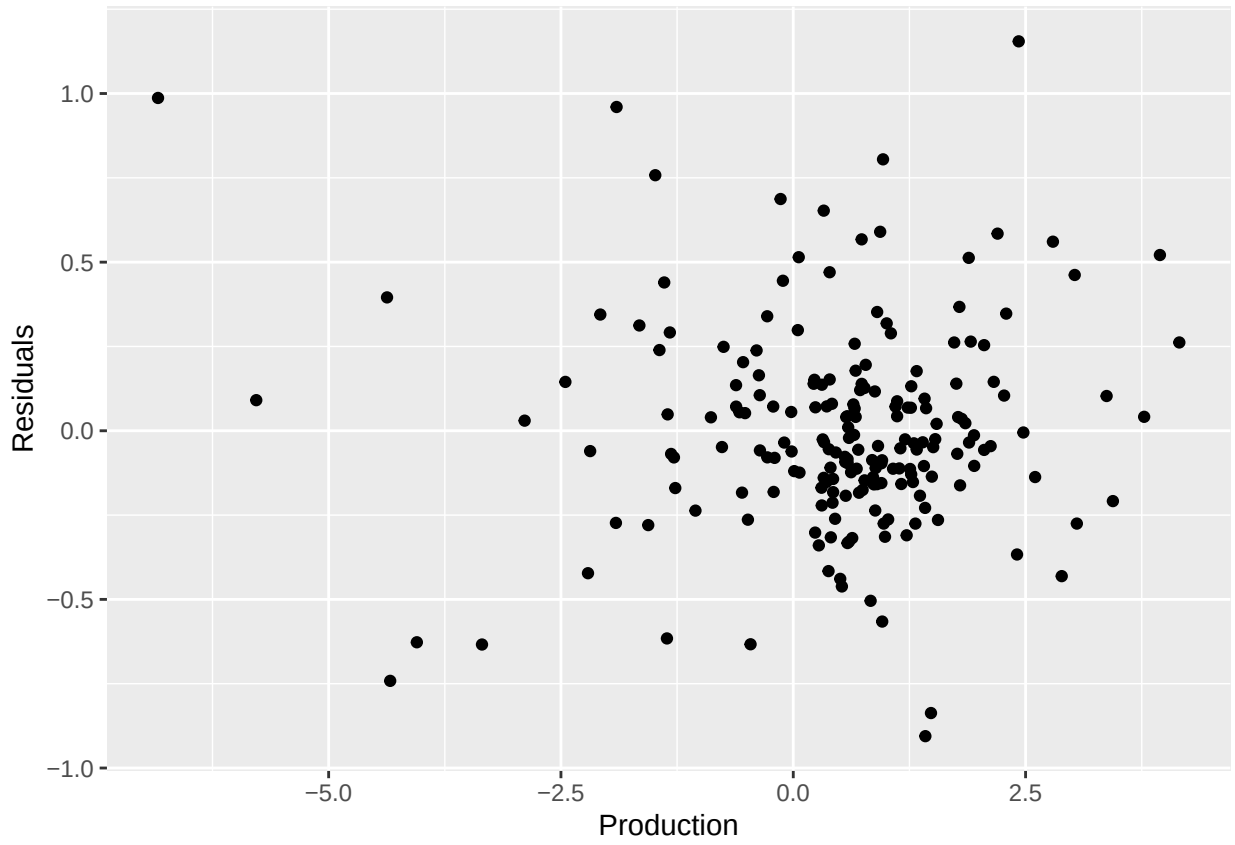
```
## # A tibble: 1 x 3
##   .model lb_stat lb_pvalue
##   <chr>    <dbl>     <dbl>
## 1 tslm      18.9   0.00204
```

Residual plots against predictors

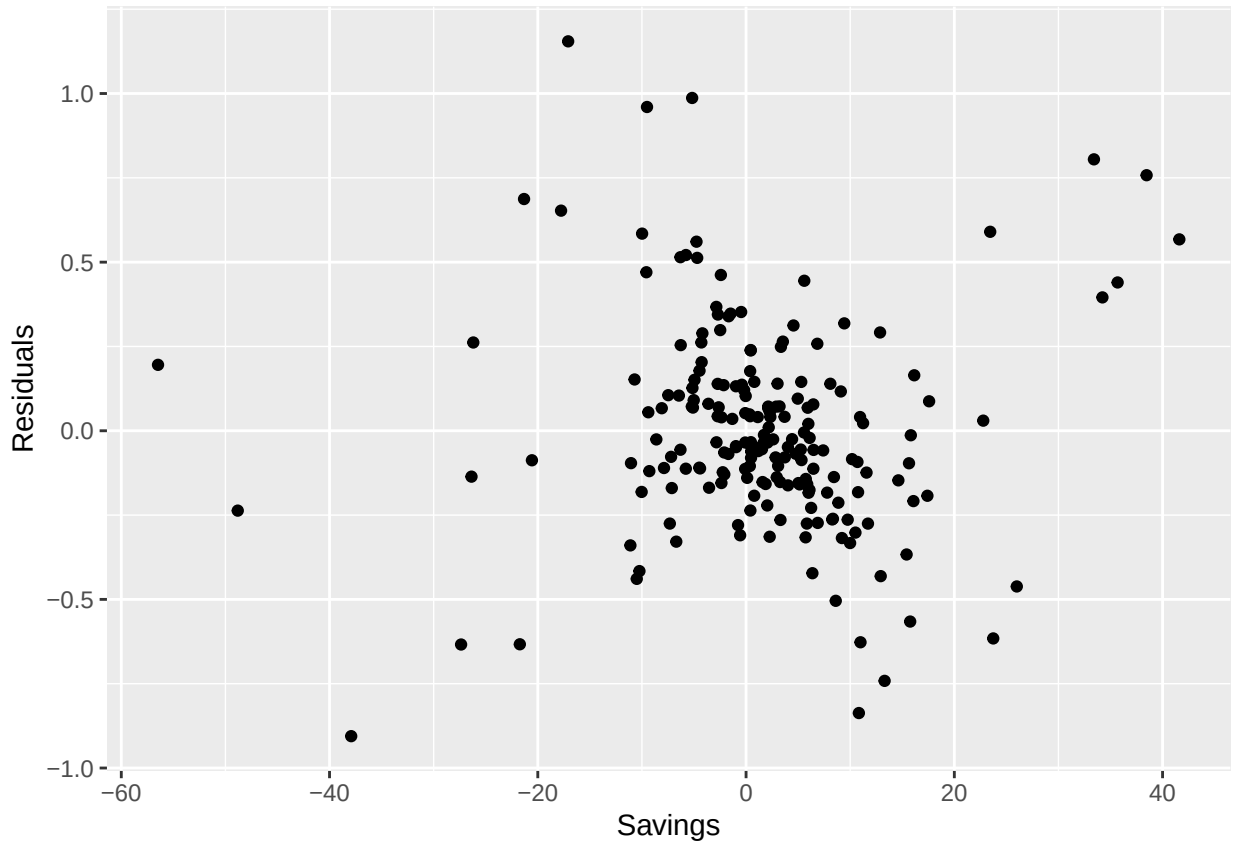
```
df <- left_join(us_change, residuals(fit.consMR), by = "Quarter")
p1 <- ggplot(df, aes(x = Income, y = .resid)) +
  geom_point() + labs(y = "Residuals")
p2 <- ggplot(df, aes(x = Production, y = .resid)) +
  geom_point() + labs(y = "Residuals")
p3 <- ggplot(df, aes(x = Savings, y = .resid)) +
  geom_point() + labs(y = "Residuals")
p4 <- ggplot(df, aes(x = Unemployment, y = .resid)) +
  geom_point() + labs(y = "Residuals")
p1
```



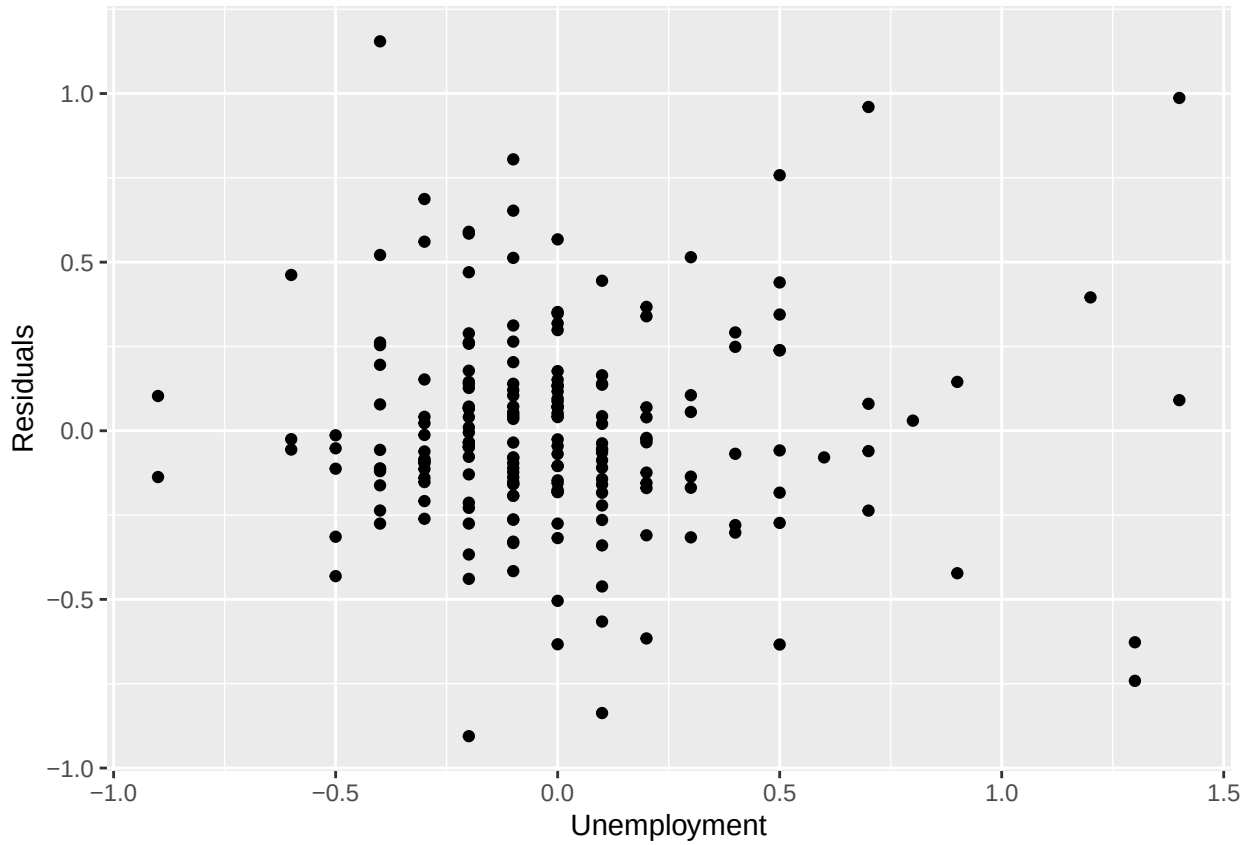

```
p2
```




```
p3
```

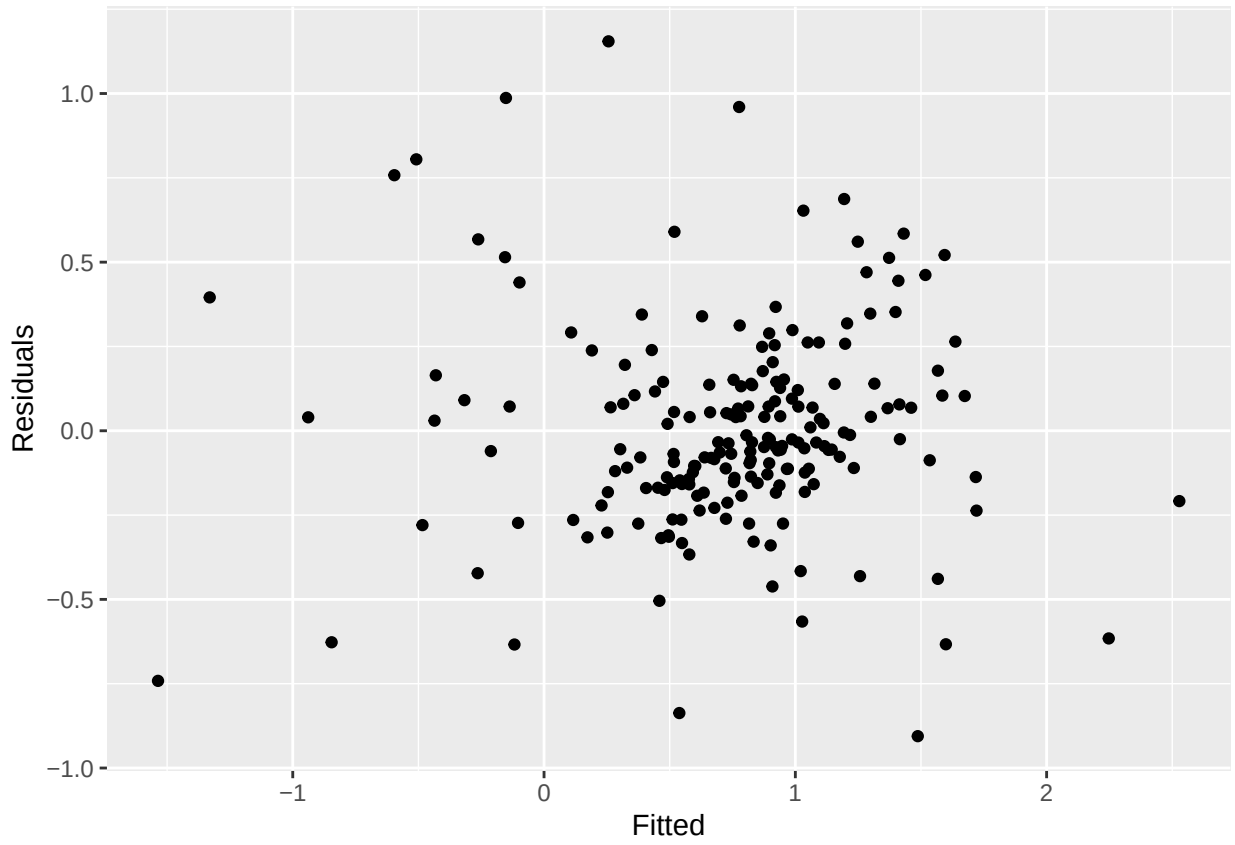



```
p4
```



Residual plots against fitted values

```
augment(fit.consMR) %>%
  ggplot(aes(x = .fitted, y = .resid)) +
  geom_point() + labs(x = "Fitted", y = "Residuals")
```

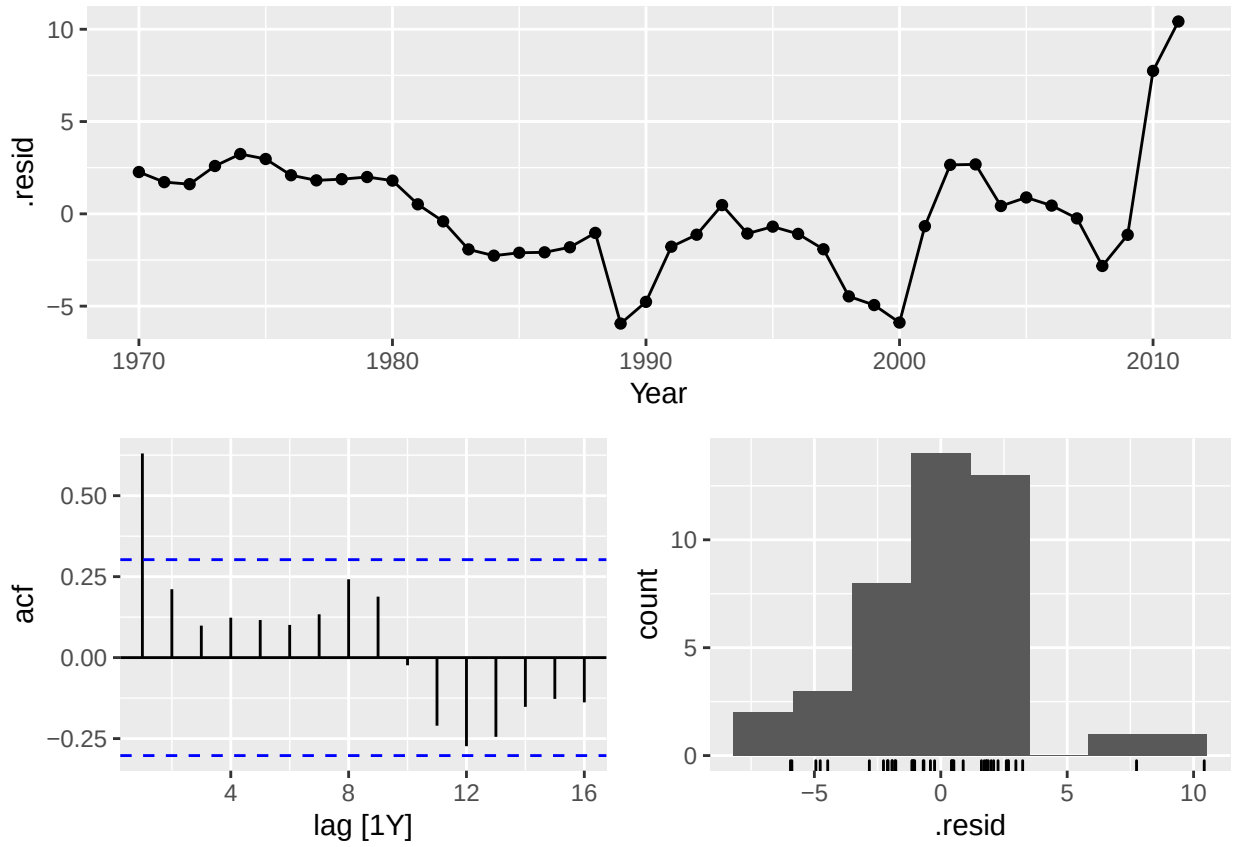


Spurious Regression

```
fit <- aus_airpassengers %>%
  filter(Year <= 2011) %>%
  left_join(guinea_rice, by = "Year") %>%
  model(TSLM(Passengers ~ Production))
report(fit)
```

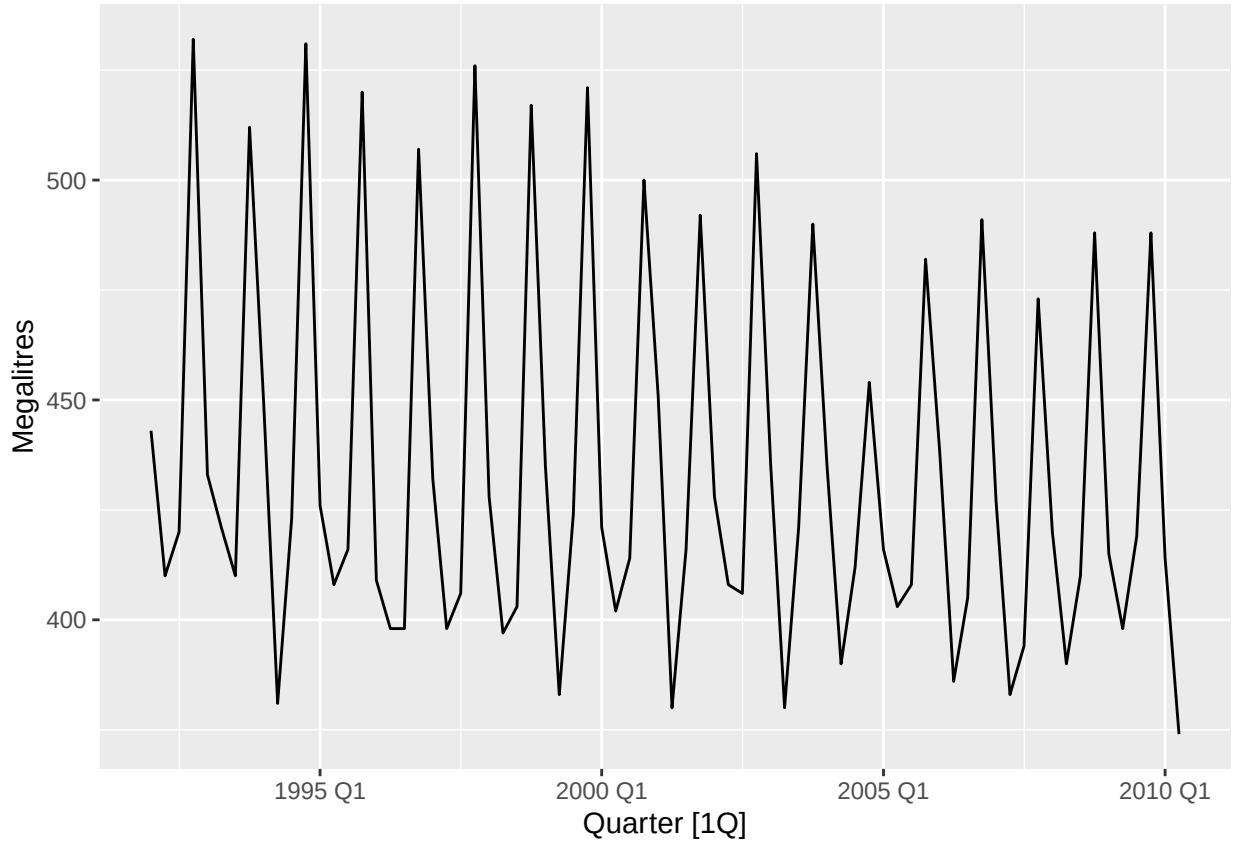
```
## Series: Passengers
## Model: TSLM
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5.9448 -1.8917 -0.3272  1.8620 10.4210
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   -7.493      1.203  -6.229 2.25e-07 ***
## Production    40.288      1.337  30.135  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.239 on 40 degrees of freedom
## Multiple R-squared: 0.9578,  Adjusted R-squared: 0.9568
## F-statistic: 908.1 on 1 and 40 DF, p-value: < 2.22e-16
```

```
fit %>% gg_tsresiduals()
```



Some Userful Predictions ### Trend, Dummy Variable, Seasonal Dummy Variable

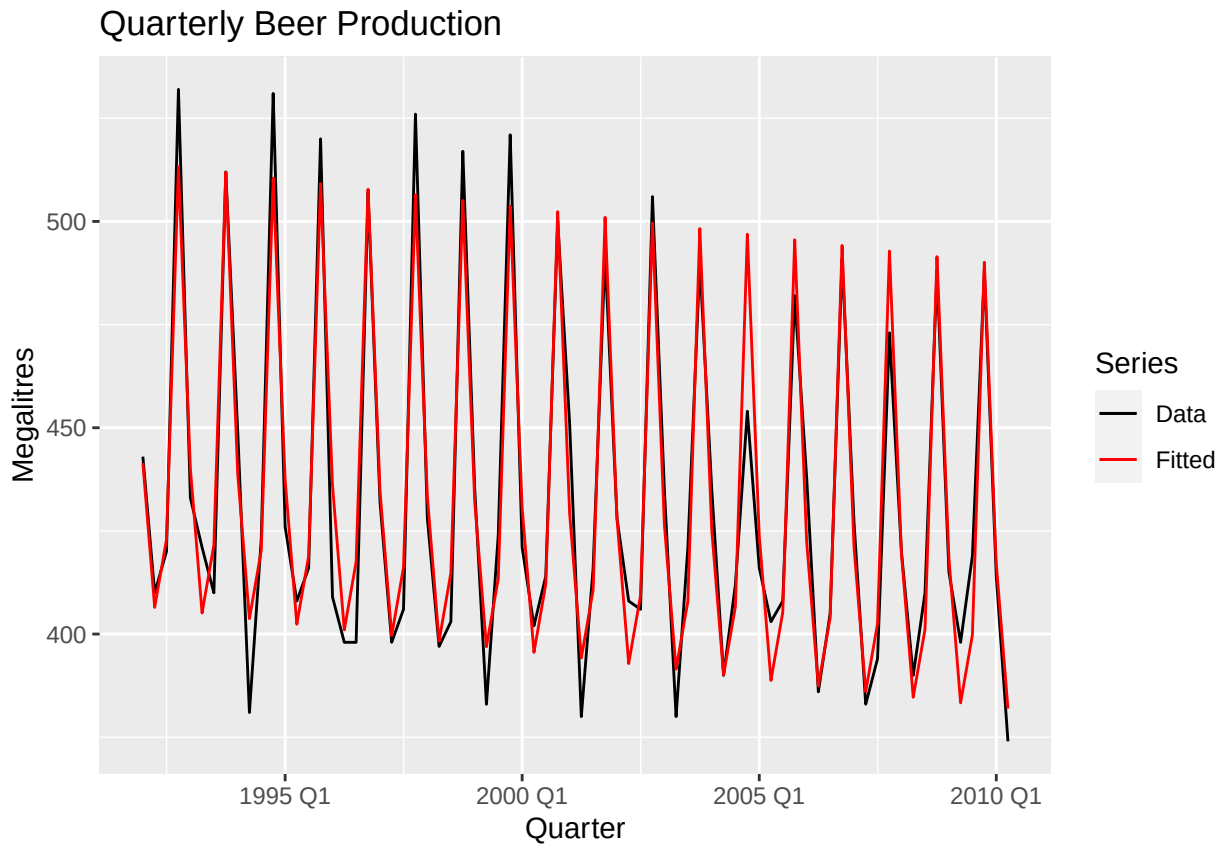
```
recent_production <- aus_production %>%
  filter(year(Quarter) >= 1992)
recent_production %>%
  autoplot(Beer) +
  labs(y = "Megalitres")
```



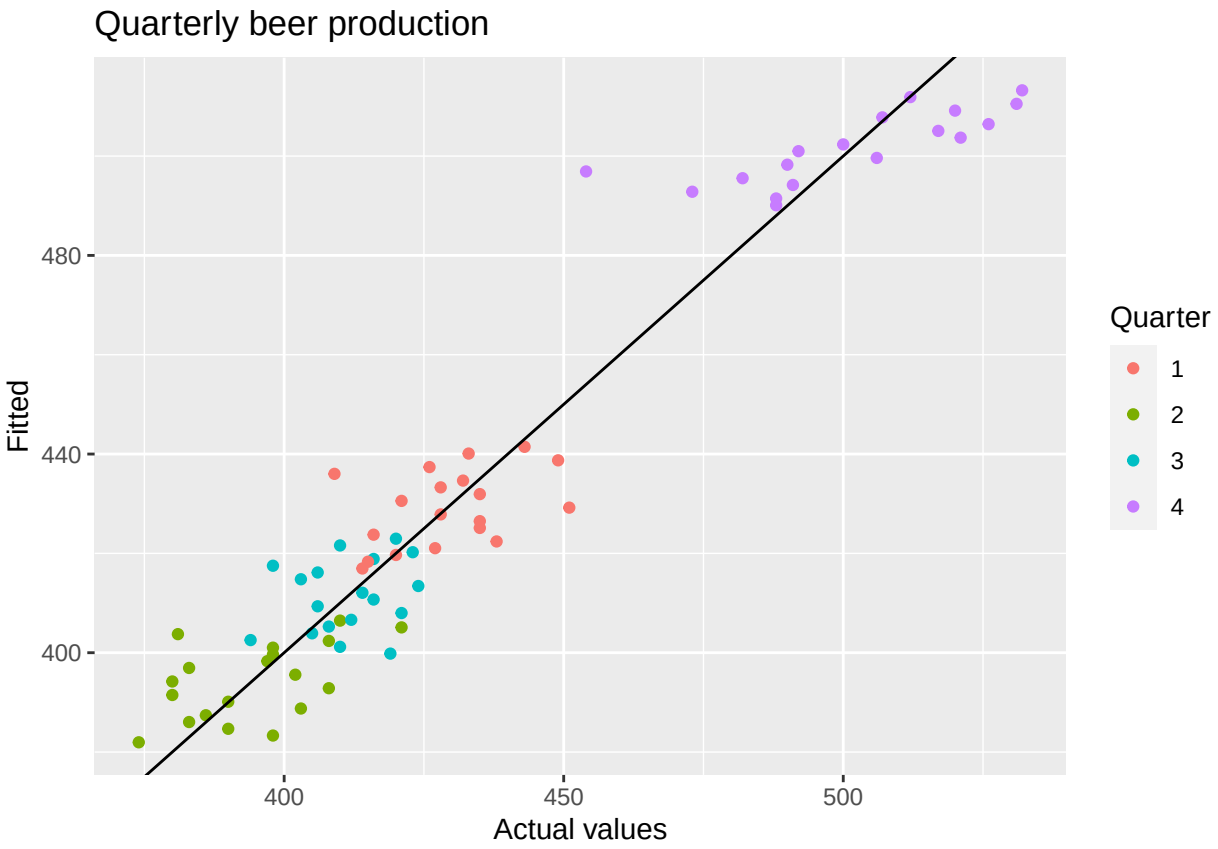

```
fit_beer <- recent_production %>%
  model(TSLM(Beer ~ trend() + season()))
report(fit_beer)
```

```
## Series: Beer
## Model: TSLM
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -42.9029  -7.5995  -0.4594   7.9908  21.7895
##
## 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 ***
## season()year2 -34.65973    3.96832  -8.734 9.10e-13 ***
## season()year3 -17.82164    4.02249  -4.430 3.45e-05 ***
## season()year4  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: < 2.22e-16
```

```
augment(fit_beer) %>%
  ggplot(aes(x = Quarter)) +
  geom_line(aes(y = Beer, colour = "Data")) +
  geom_line(aes(y = .fitted, colour = "Fitted")) +
  scale_color_manual(values = c(Data = "black", Fitted = "red")) +
  labs(y = "Megalitres", title = "Quarterly Beer Production") +
  guides(colour = guide_legend(title = "Series"))
```




```
augment(fit_beer) %>%
  ggplot(aes(x = Beer, y = .fitted, colour = factor(quarter(Quarter)))) +
  geom_point() +
  labs(y = "Fitted", x = "Actual values",
       title = "Quarterly beer production") +
  geom_abline(intercept = 0, slope = 1) +
  guides(colour = guide_legend(title = "Quarter"))
```



Fourier Transform

```
fourier_beer <- recent_production %>%
  model(TSLM(Beer ~ trend() + fourier(K = 2)))
report(fourier_beer)
```

```
## Series: Beer
## Model: TSLM
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -42.9029  -7.5995  -0.4594   7.9908  21.7895
##
## 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(K = 2)C1_4   8.91082    2.01125   4.430 3.45e-05 ***
## fourier(K = 2)S1_4 -53.72807    2.01125 -26.714  < 2e-16 ***
## fourier(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: < 2.22e-16
```

Selecting predictors

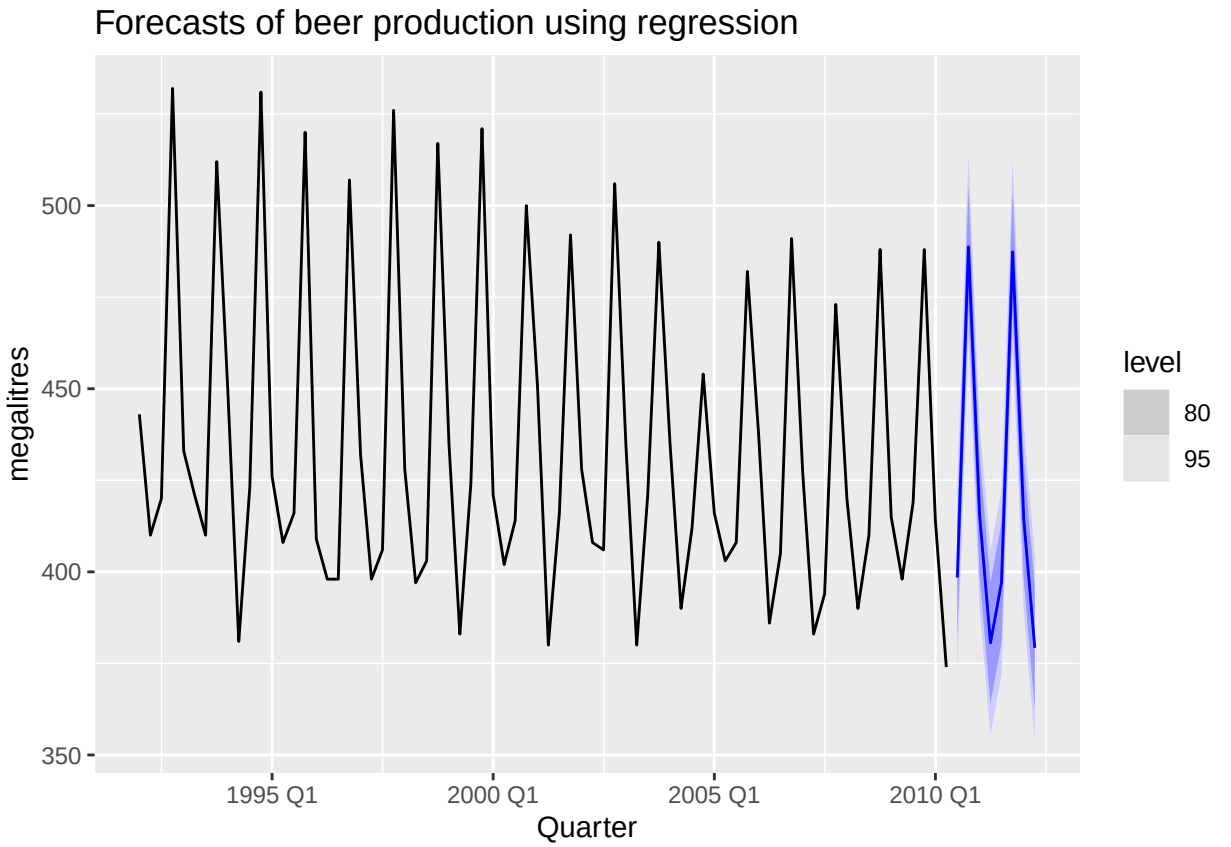
```
glance(fit.consMR) %>% select(adj_r_squared, CV, AIC, AICc, BIC)
```

```
## # A tibble: 1 x 5
##   adj_r_squared    CV   AIC  AICc   BIC
##           <dbl> <dbl> <dbl> <dbl> <dbl>
## 1         0.763 0.104 -457. -456. -437.
```

Forecasting with regression

Ex-ante vs Ex-posts forecast

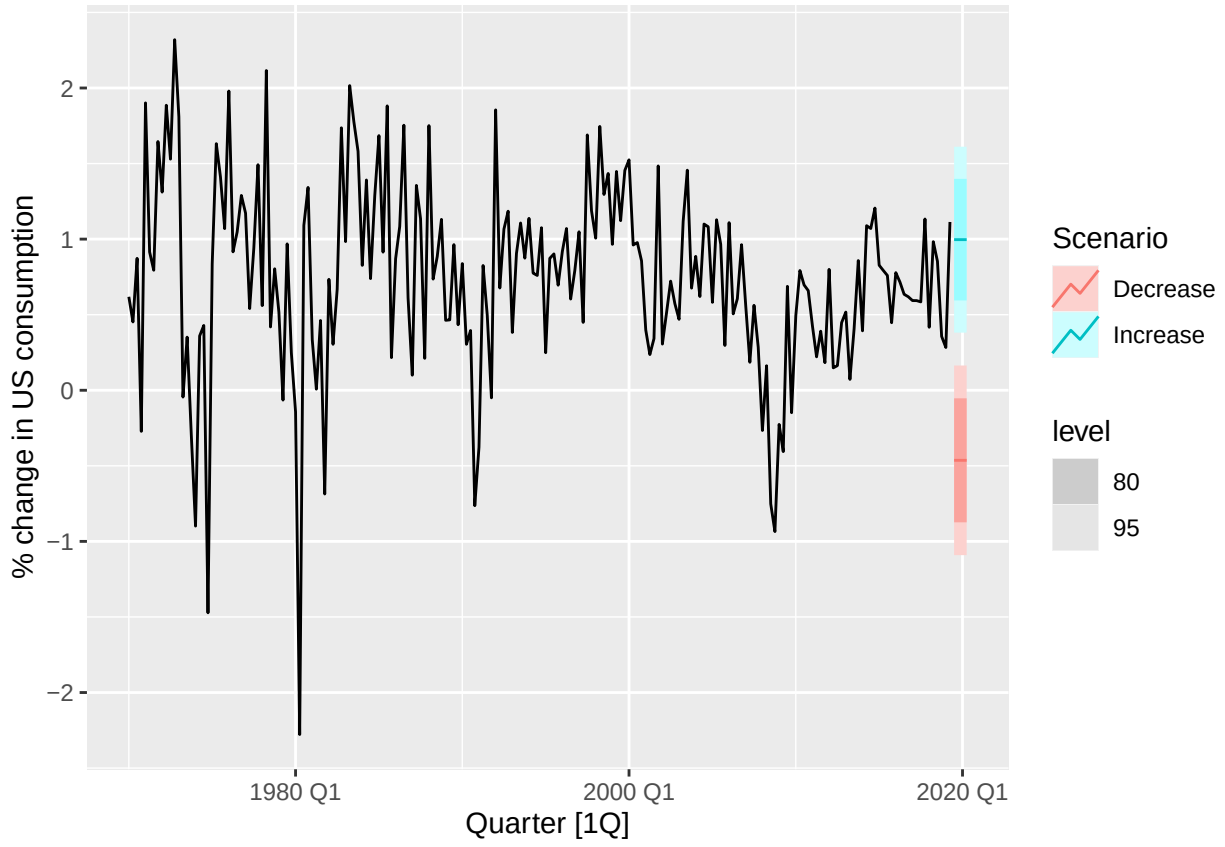
```
recent_production <- aus_production %>% filter(year(Quarter) >= 1992)
fit_beer <- recent_production %>%
  model(TSLM(Beer ~ trend() + season()))
fc_beer <- forecast(fit_beer)
fc_beer %>%
  autoplot(recent_production) +
  labs(title = "Forecasts of beer production using regression",
       y = "megalitres")
```



Scenario-based forecast

```
fit_consBest <- us_change %>%
  model(lm = TSLM(Consumption ~ Income + Savings + Unemployment))

future_scenarios <- scenarios(
  Increase = new_data(us_change, 4) %>% mutate(Income=1, Savings=0.5, Unemployment=0),
  Decrease = new_data(us_change, 4) %>% mutate(Income=-1, Savings=-0.5, Unemployment=0),
  names_to = "Scenario")

fc <- forecast(fit_consBest, new_data = future_scenarios)
```

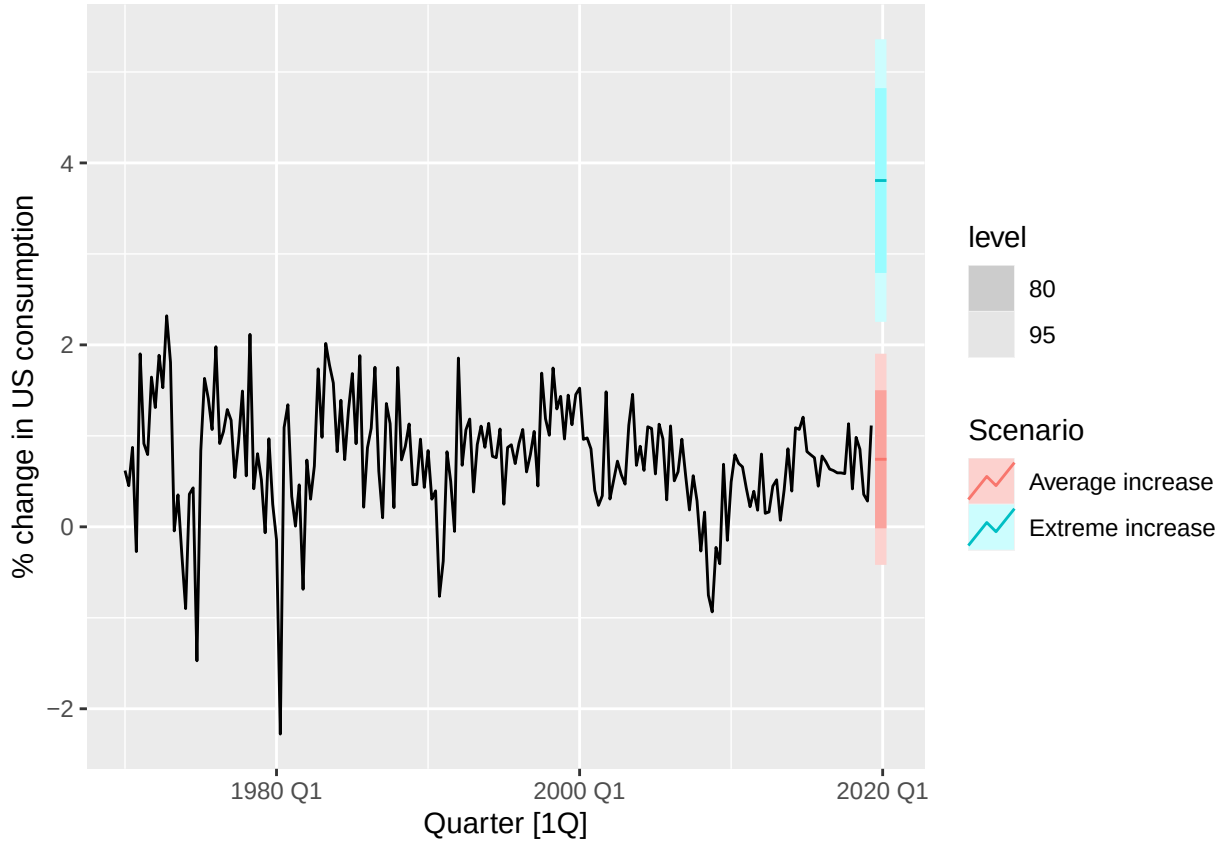
```
us_change %>%
  autoplot(Consumption) +
  autolayer(fc) +
  labs(y = "% change in US consumption")
```



Prediction Intervals

```
fit_cons <- us_change %>%
  model(TSLM(Consumption ~ Income))
new_cons <- scenarios(
  "Average increase" = new_data(us_change, 4) %>% mutate(Income = mean(us_change$Income)),
  "Extreme increase" = new_data(us_change, 4) %>% mutate(Income = 12),
  names_to = "Scenario"
```

```
)
fcast <- forecast(fit_cons, new_cons)

us_change %>%
  autoplot(Consumption) +
  autolayer(fcast) +
  labs(y = "% change in US consumption")
```



Nonlinear Regression

```
boston_men <- boston_marathon %>%
  filter(Year >= 1924) %>%
  filter(Event == "Men's open division") %>%
  mutate(Minutes = as.numeric(Time)/60)

boston_men
```

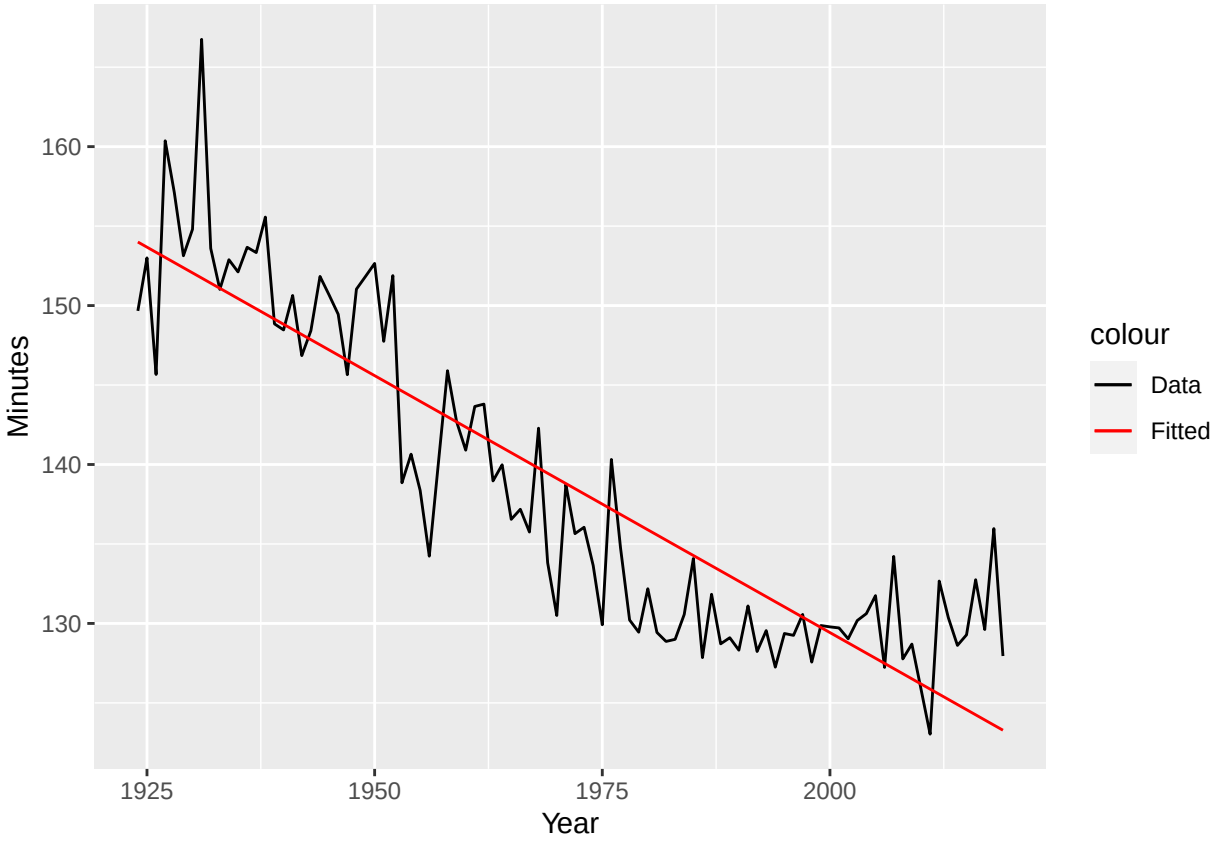
```
## # A tsibble: 96 x 6 [1Y]
## # Key:       Event [1]
##    Event               Year Champion               Country      Time     Minutes
##    <fct>              <int> <chr>                  <chr>        <time>     <dbl>
##  1 Men's open divisi~  1924 Clarence H. DeMar      United Stat~ 02:29:40    150.
##  2 Men's open divisi~  1925 Charles L. (Chuck) Me~ United Stat~ 02:33:00    153
```

```
##  3 Men's open divisi~  1926 John C. Miles          Canada       02:25:40    146.
##  4 Men's open divisi~  1927 Clarence H. DeMar      United Stat~ 02:40:22    160.
##  5 Men's open divisi~  1928 Clarence H. DeMar      United Stat~ 02:37:07    157.
##  6 Men's open divisi~  1929 John C. Miles          Canada       02:33:08    153.
##  7 Men's open divisi~  1930 Clarence H. DeMar      United Stat~ 02:34:48    155.
##  8 Men's open divisi~  1931 James P. Henigan       United Stat~ 02:46:45    167.
##  9 Men's open divisi~  1932 Paul de Bruyn          Germany      02:33:36    154.
## 10 Men's open divisi~  1933 Leslie S. Pawson       United Stat~ 02:31:01    151.
## # ... with 86 more rows
```

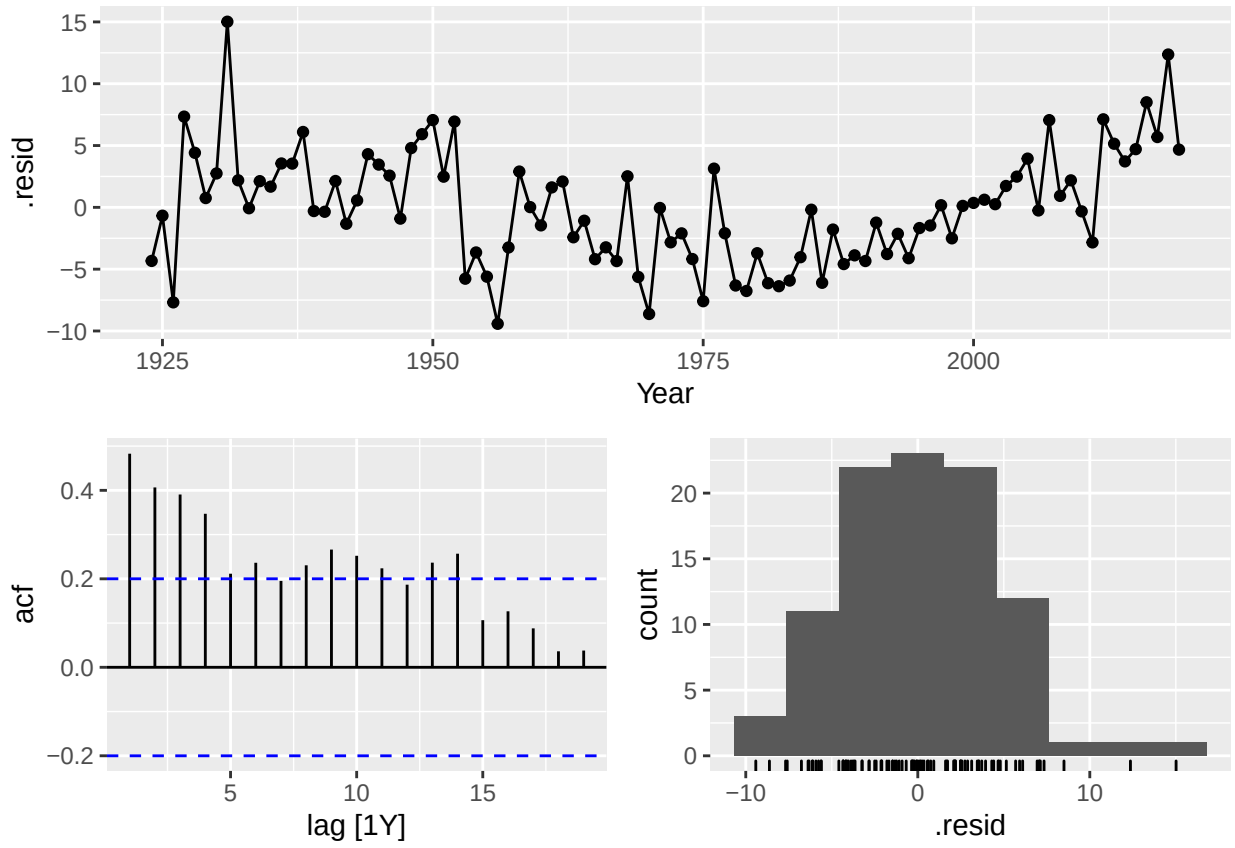
```
fit_boston_men <- boston_men %>%
  model(TSLM(Minutes ~ trend()))
report(fit_boston_men)
```

```
## Series: Minutes
## Model: TSLM
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -9.4178 -3.6628 -0.1312  2.7806 15.0157
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 154.32094    0.94507  163.29   <2e-16 ***
## trend()      -0.32333    0.01692  -19.11   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.594 on 94 degrees of freedom
## Multiple R-squared: 0.7953,  Adjusted R-squared: 0.7931
## F-statistic: 365.2 on 1 and 94 DF, p-value: < 2.22e-16
```

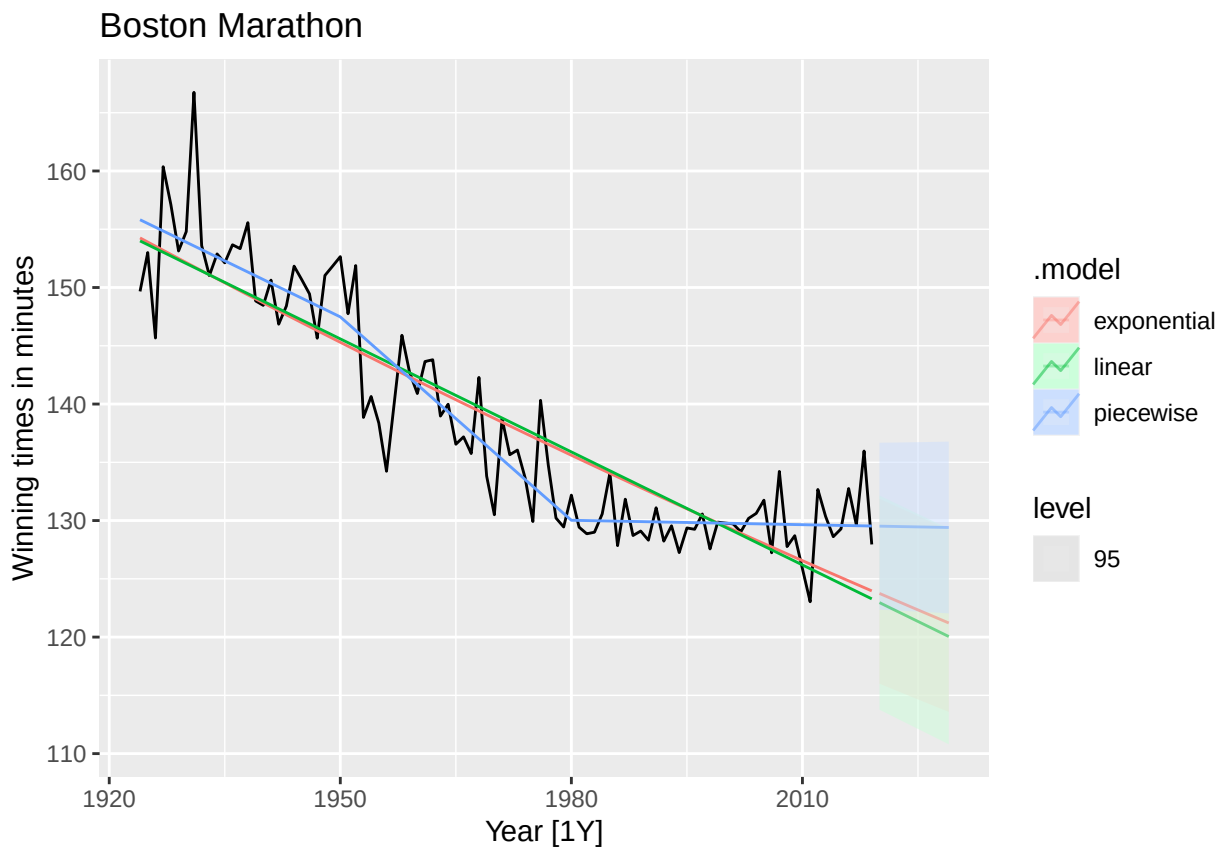
```
augment(fit_boston_men) %>%
  ggplot(aes(x = Year)) +
  geom_line(aes(y = Minutes, colour = "Data")) +
  geom_line(aes(y = .fitted, colour = "Fitted")) +
  scale_color_manual(values = c(Data = "black", Fitted = "red"))
```




```
fit_boston_men %>% gg_tsresiduals()
```




```
fit_trends <- boston_men %>%
  model(
    linear = TSLM(Minutes ~ trend()),
    exponential = TSLM(log(Minutes) ~ trend()),
    piecewise = TSLM(Minutes ~ trend(knots = c(1950, 1980)))
  )
fc_trends <- fit_trends %>% forecast(h = 10)

boston_men %>%
  autoplot(Minutes) +
  geom_line(aes(y = .fitted, colour = .model), data = fitted(fit_trends)) +
  autolayer(fc_trends, alpha = 0.5, level = 95) +
  labs(y = "Winning times in minutes",
       title = "Boston Marathon")
```



Exercise

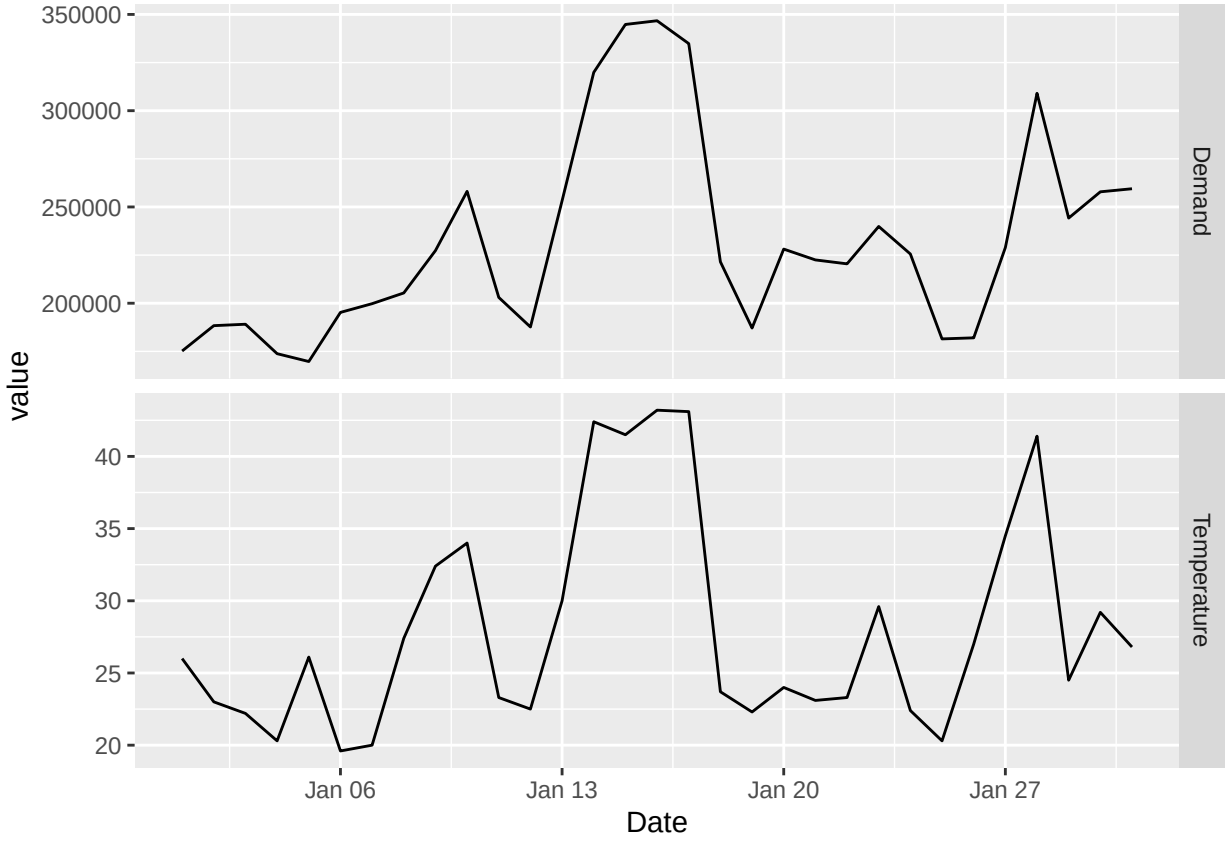
Question 1

```
jan14_vic_elec <- vic_elec %>%
  filter(yearmonth(Time) == yearmonth("2014 Jan")) %>%
  index_by(Date = as_date(Time)) %>%
  summarise(Demand = sum(Demand), Temperature = max(Temperature))

jan14_vic_elec
```

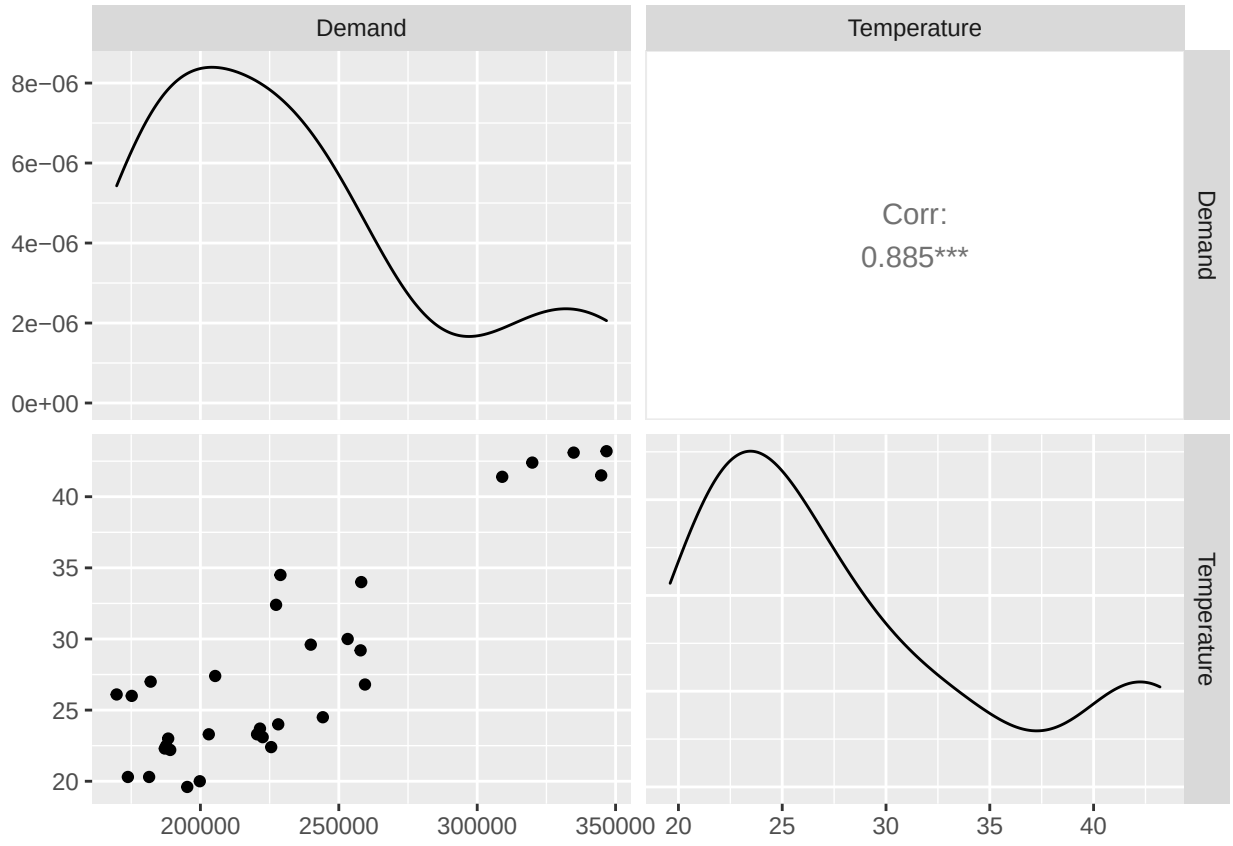
```
## # A tsibble: 31 x 3 [1D]
##    Date        Demand Temperature
##    <date>       <dbl>       <dbl>
##  1 2014-01-01 175185.        26
##  2 2014-01-02 188351.        23
##  3 2014-01-03 189086.        22.2
##  4 2014-01-04 173798.        20.3
##  5 2014-01-05 169733.        26.1
##  6 2014-01-06 195241.        19.6
##  7 2014-01-07 199770.        20
##  8 2014-01-08 205339.        27.4
##  9 2014-01-09 227334.        32.4
```

```
## 10 2014-01-10 258111.        34
## # ... with 21 more rows
```

```
jan14_vic_elec %>% pivot_longer(c(Demand, Temperature), names_to = "Series") %>% ggplot(aes(x = Date, y
```

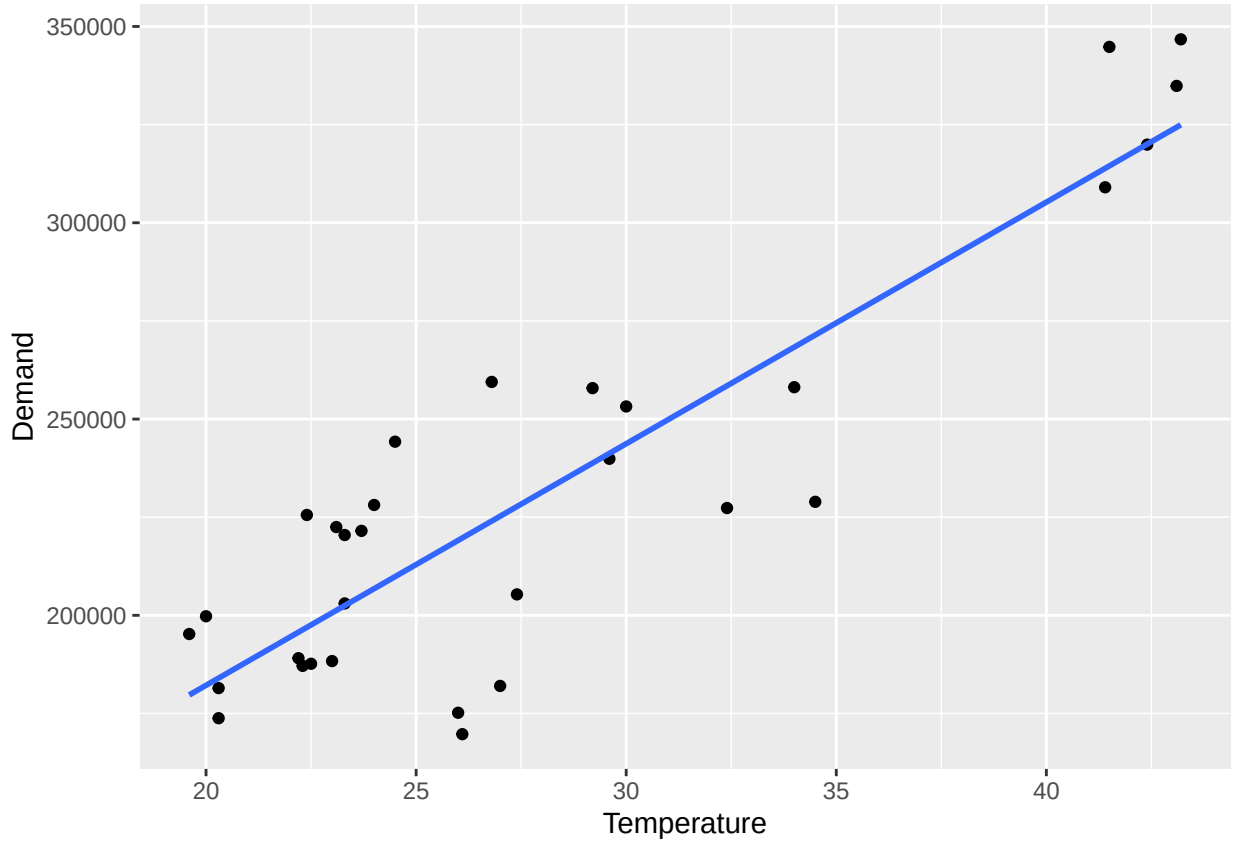



```
jan14_vic_elec %>% GGally::ggpairs(columns = 2:3)
```




```
jan14_vic_elec %>%
  ggplot(aes(x = Temperature, y = Demand)) +
  geom_point() +
  geom_smooth(method = "lm", se = FALSE)
```

```
## `geom_smooth()` using formula 'y ~ x'
```



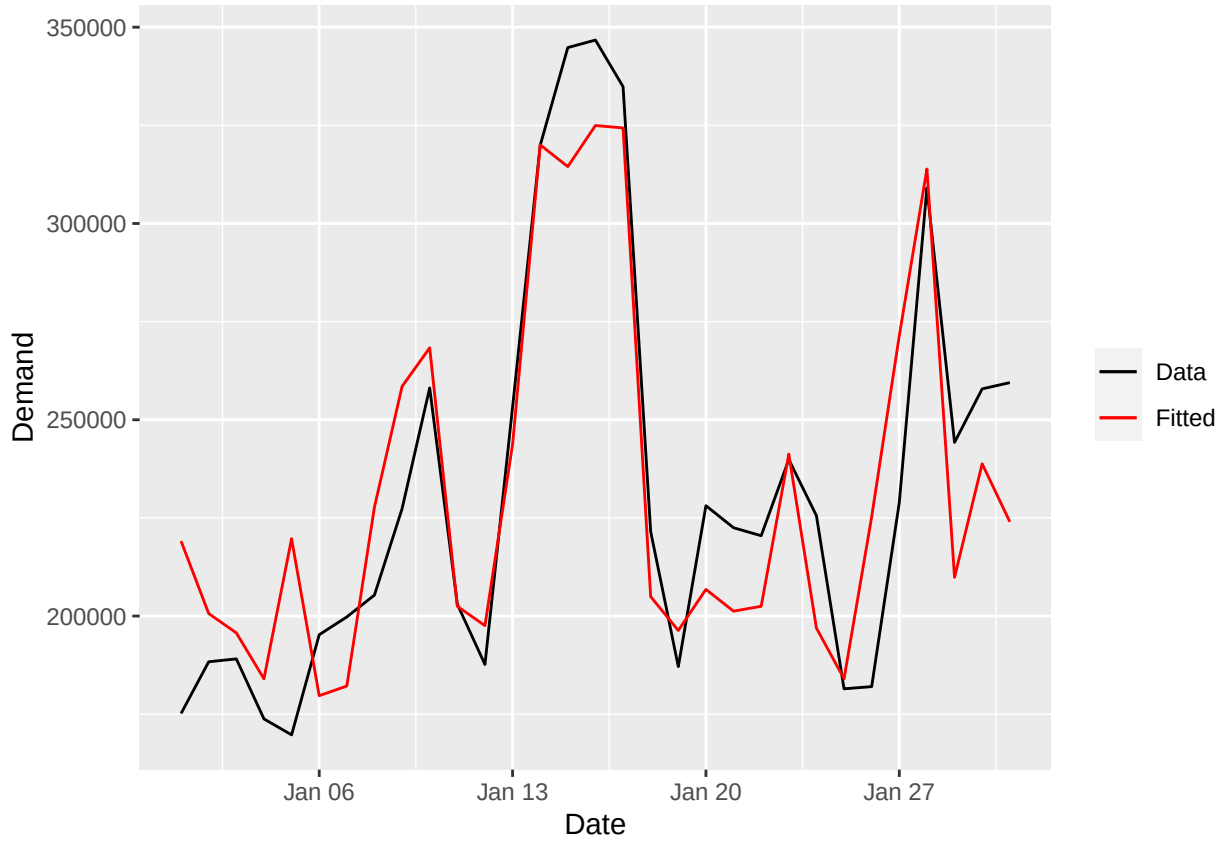

```
jan14_vic_elec %>%
  model(TSLM(Demand ~ Temperature)) %>%
  report()
```

```
## Series: Demand
## Model: TSLM
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -49978.2 -10218.9   -121.3  18533.2  35440.6
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  59083.9    17424.8   3.391  0.00203 **
## Temperature   6154.3      601.3  10.235 3.89e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 24540 on 29 degrees of freedom
## Multiple R-squared: 0.7832,  Adjusted R-squared: 0.7757
## F-statistic: 104.7 on 1 and 29 DF, p-value: 3.8897e-11
```

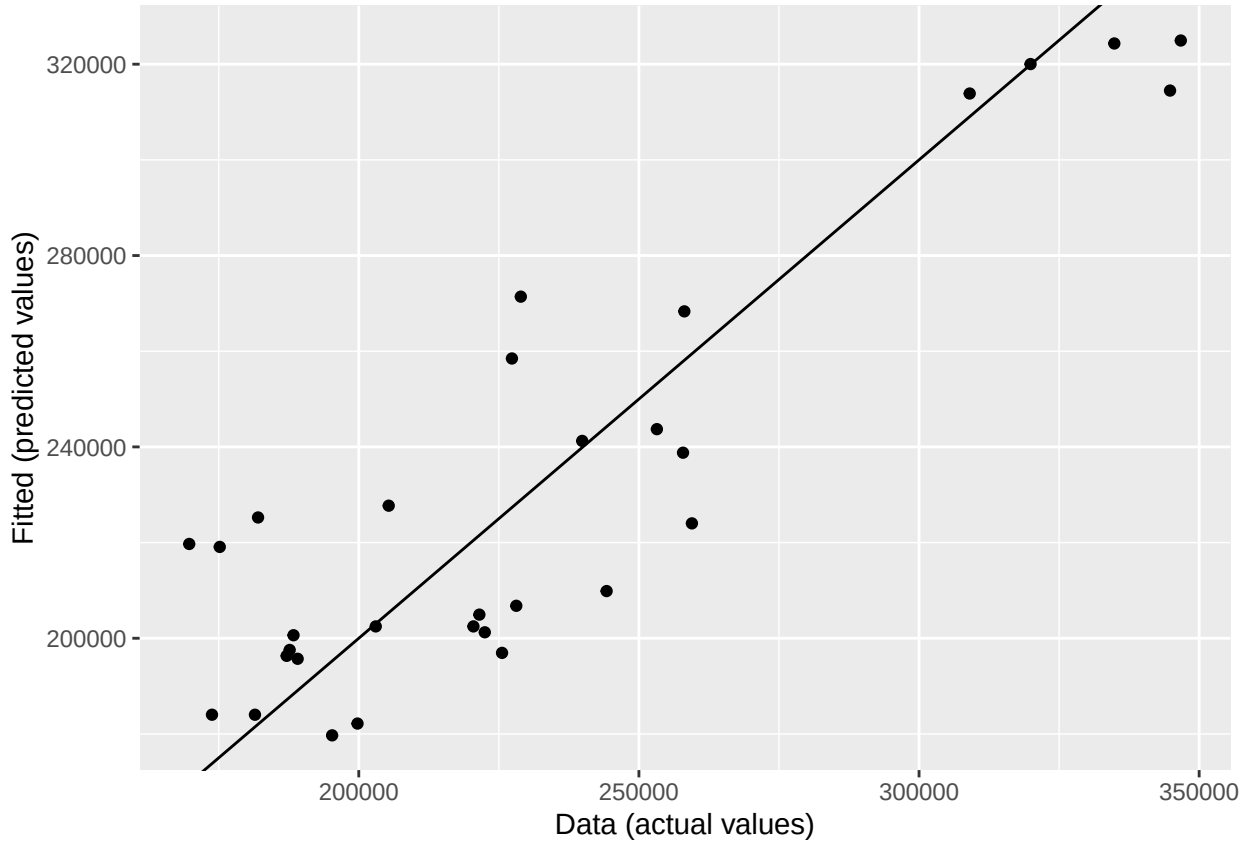
```
fit.consMR <- jan14_vic_elec %>%
  model(tslm = TSLM(Demand ~ Temperature))
report(fit.consMR)
```

```
## Series: Demand
## Model: TSLM
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -49978.2 -10218.9   -121.3  18533.2  35440.6
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  59083.9    17424.8   3.391  0.00203 **
## Temperature   6154.3      601.3  10.235 3.89e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 24540 on 29 degrees of freedom
## Multiple R-squared: 0.7832,  Adjusted R-squared: 0.7757
## F-statistic: 104.7 on 1 and 29 DF, p-value: 3.8897e-11
```

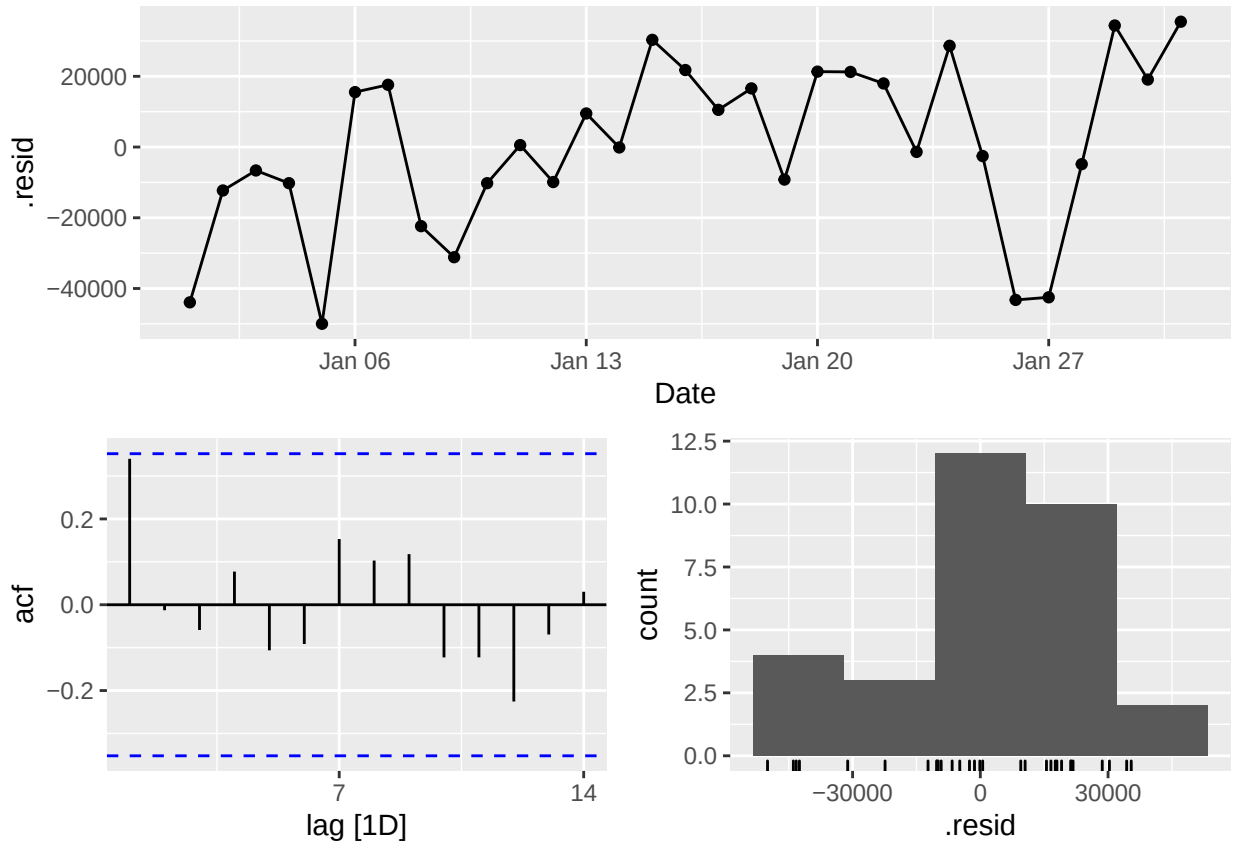
```
augment(fit.consMR) %>%
  ggplot(aes(x = Date)) +
  geom_line(aes(y = Demand, colour = "Data")) +
  geom_line(aes(y = .fitted, colour = "Fitted")) +
  scale_color_manual(values=c(Data="black",Fitted="red")) +
  guides(colour = guide_legend(title = NULL))
```



```
augment(fit.consMR) %>%
  ggplot(aes(x = Demand, y = .fitted)) +
  geom_point() +
  labs(y = "Fitted (predicted values)",
       x = "Data (actual values)") +
  geom_abline(intercept = 0, slope = 1)
```



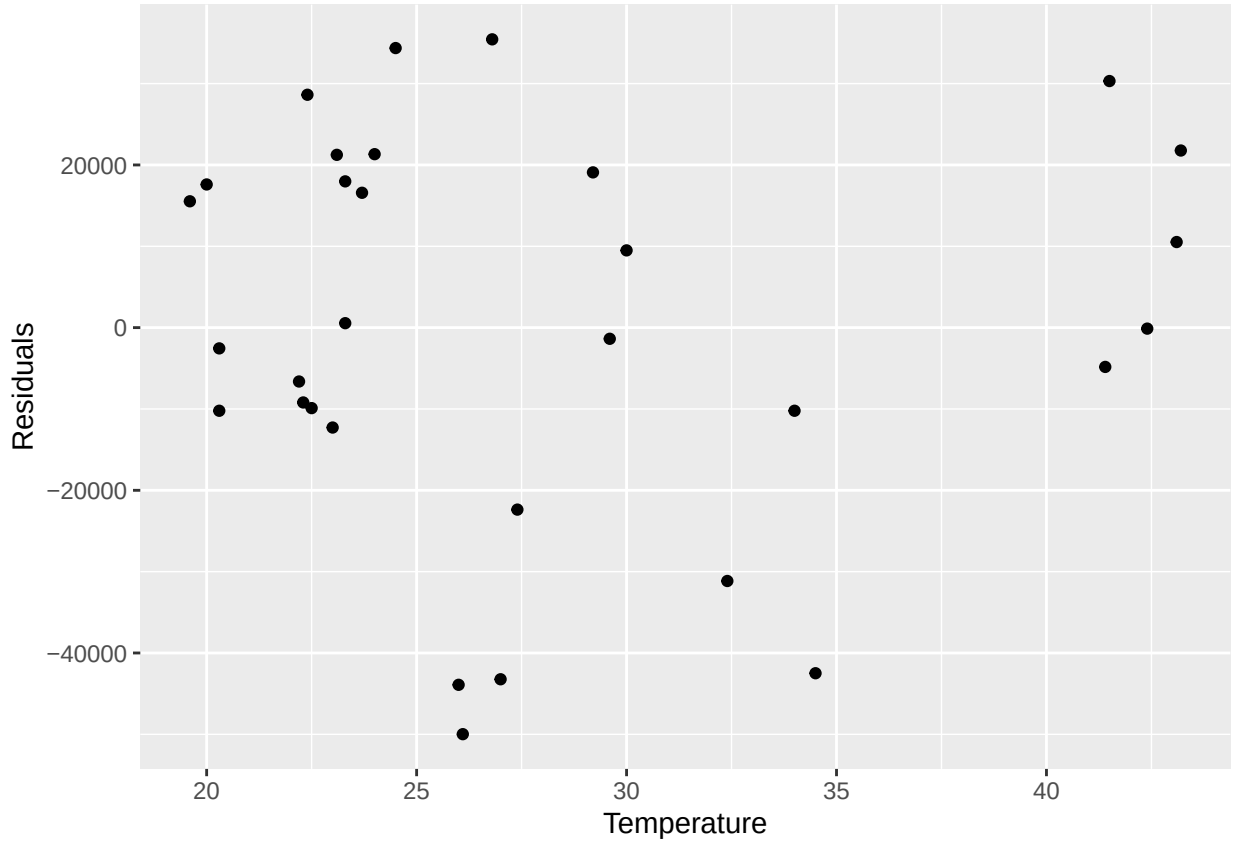

```
fit.consMR %>% gg_tsresiduals()
```



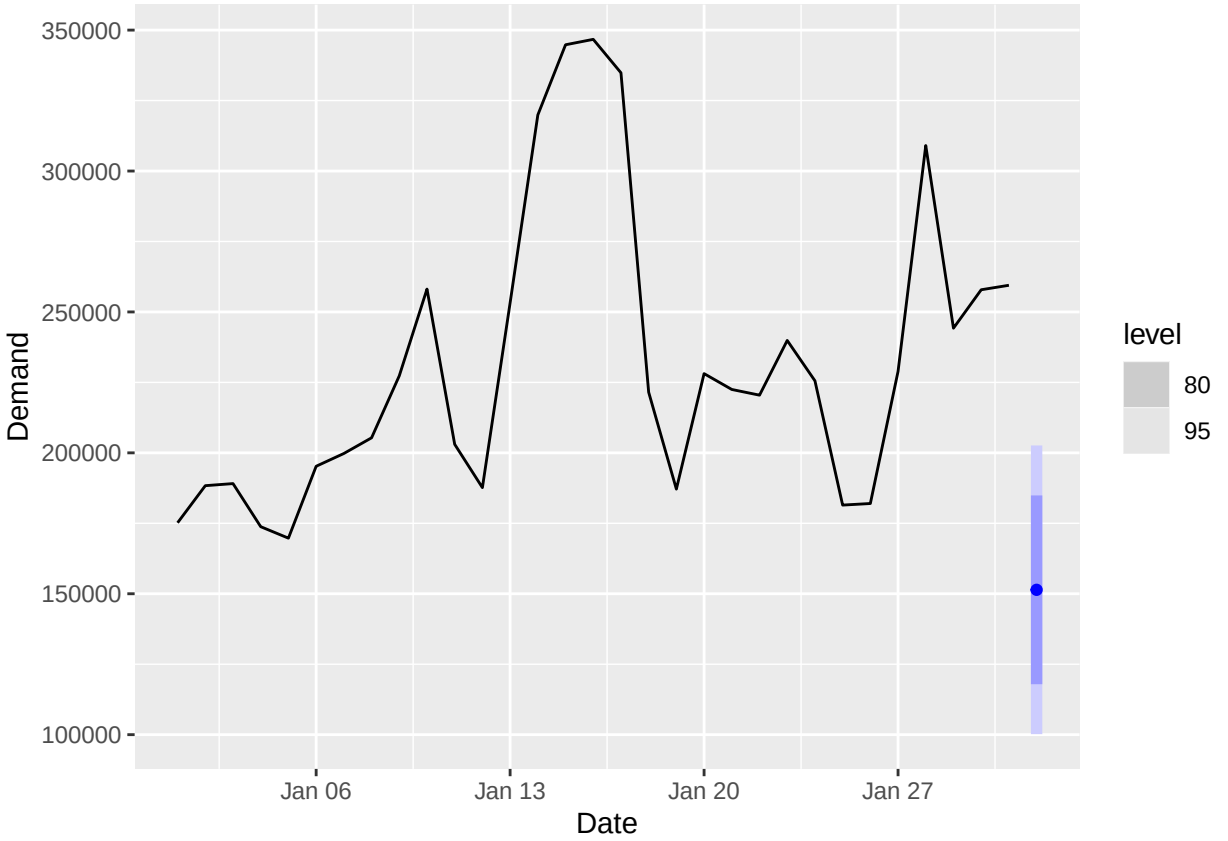

```
augment(fit.consMR) %>% features(.innov, ljung_box, lag = 10, dof = 5)
```

```
## # A tibble: 1 x 3
##   .model lb_stat lb_pvalue
##   <chr>    <dbl>     <dbl>
## 1 tslm      7.94     0.159
```

```
df <- left_join(jan14_vic_elec, residuals(fit.consMR), by = "Date")
p1 <- ggplot(df, aes(x = Temperature, y = .resid)) +
  geom_point() + labs(y = "Residuals")

p1
```



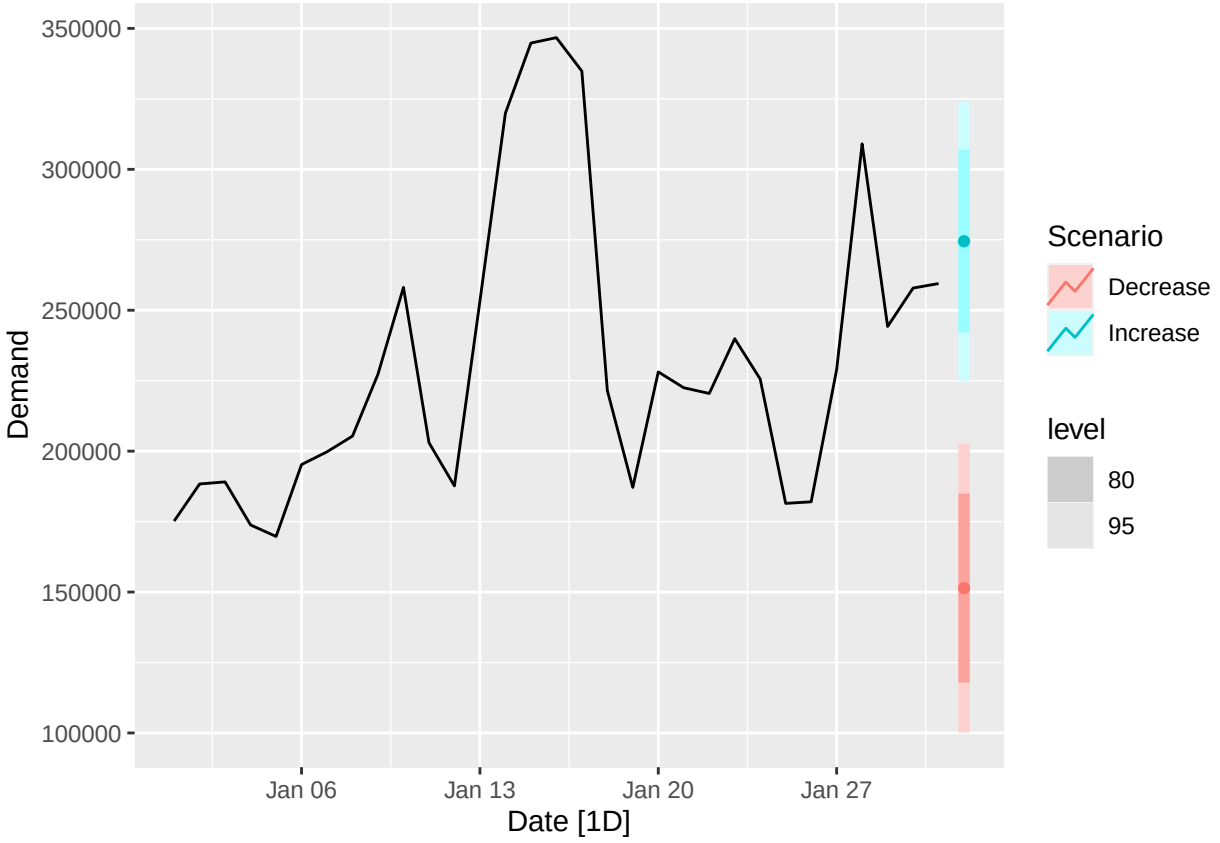

```
jan14_vic_elec %>%
  model(TSLM(Demand ~ Temperature)) %>%
  forecast(
    new_data(jan14_vic_elec, 1) %>% mutate(Temperature = 15)
  ) %>%
  autoplot(jan14_vic_elec)
```




```
new_model <- jan14_vic_elec %>% model(TSLM(Demand ~ Temperature))

future_scenarios <- scenarios(
  Increase = new_data(jan14_vic_elec, 1) %>% mutate(Temperature = 35),
  Decrease = new_data(jan14_vic_elec, 1) %>% mutate(Temperature = 15),
  names_to = "Scenario")

future_scenarios
```

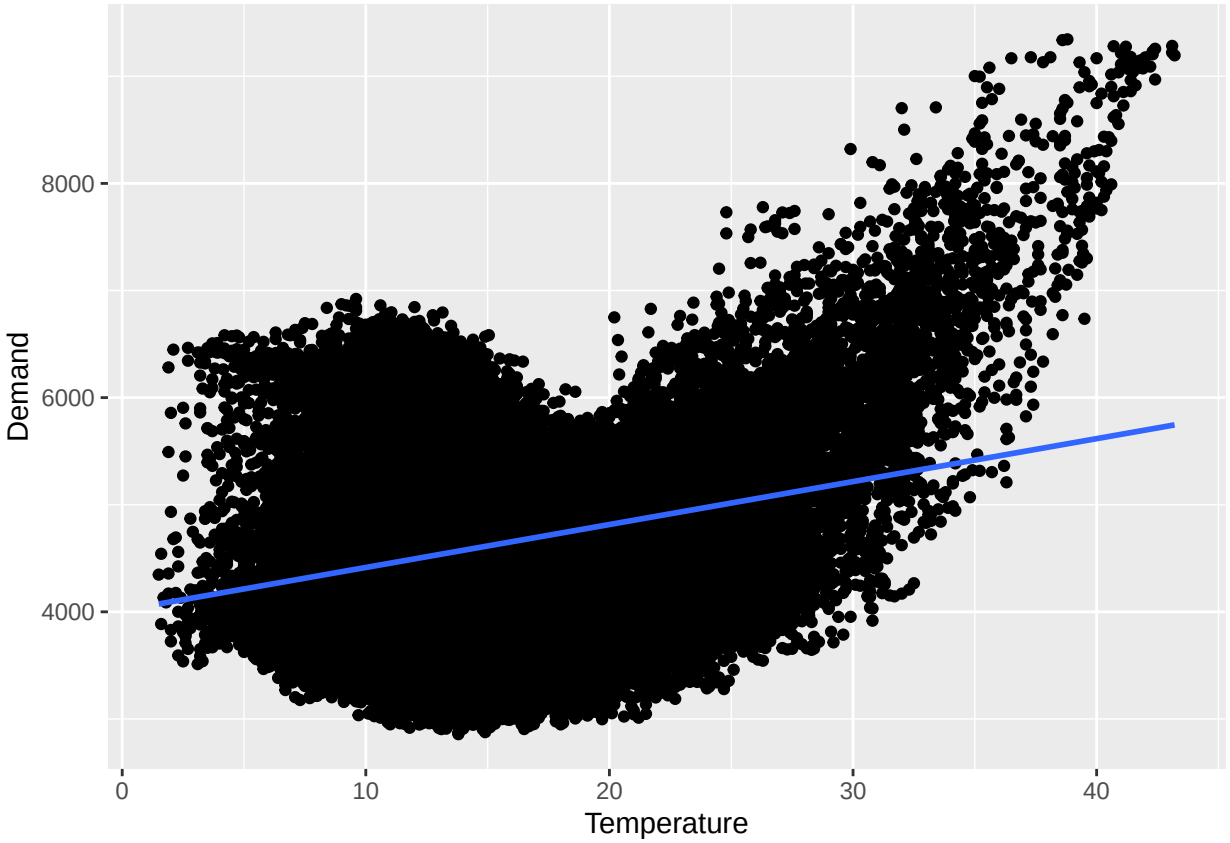
```
## $Increase
## # A tsibble: 1 x 2 [1D]
##   Date       Temperature
##   <date>           <dbl>
## 1 2014-02-01          35
##
## $Decrease
## # A tsibble: 1 x 2 [1D]
##   Date       Temperature
##   <date>           <dbl>
## 1 2014-02-01          15
##
## attr(,"names_to")
## [1] "Scenario"
```

```
fc <- forecast(new_model, new_data = future_scenarios)

jan14_vic_elec %>%
  autoplot(Demand) +
  autolayer(fc)
```




```
vic_elec %>% ggplot(aes(x = Temperature, y = Demand)) +
  geom_point() +
  geom_smooth(method = "lm", se = FALSE)
```

```
## `geom_smooth()` using formula 'y ~ x'
```



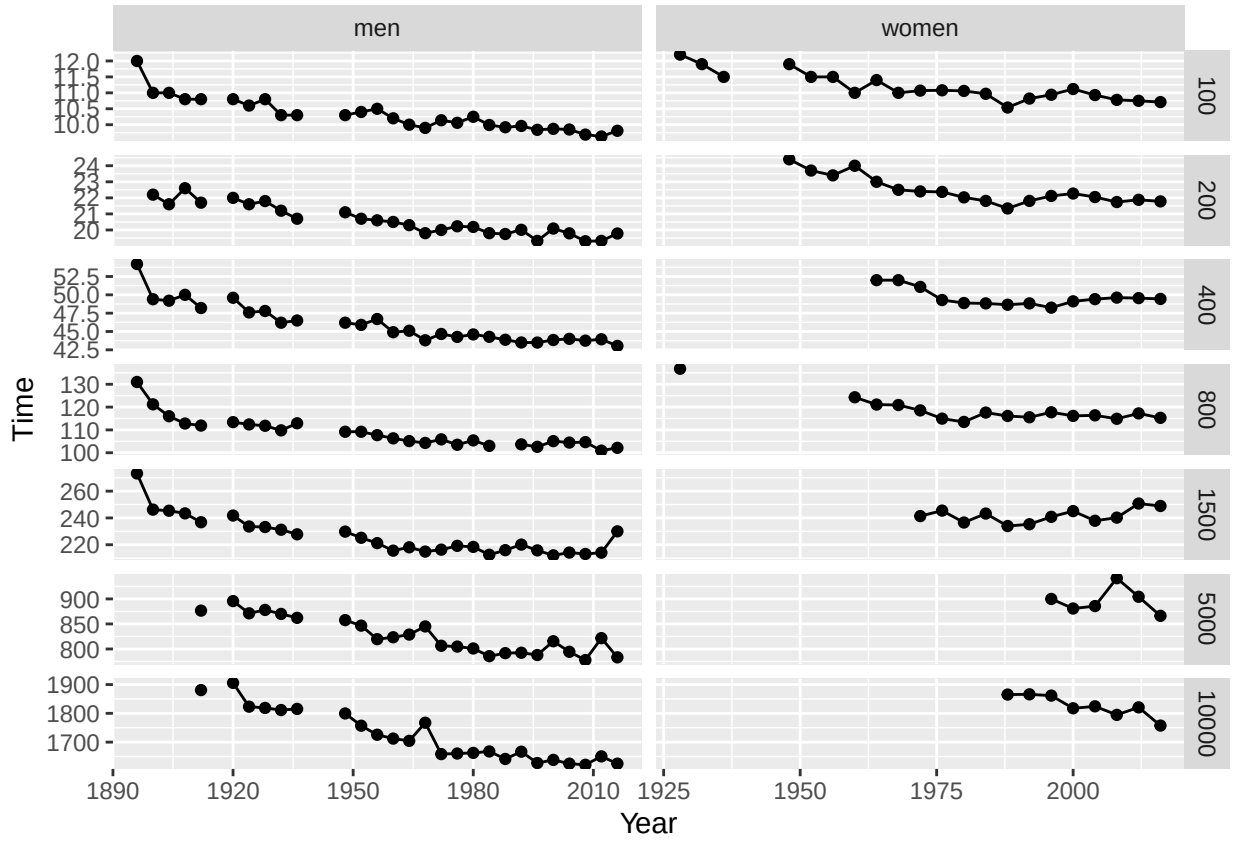
Question 2

```
olympic_running
```

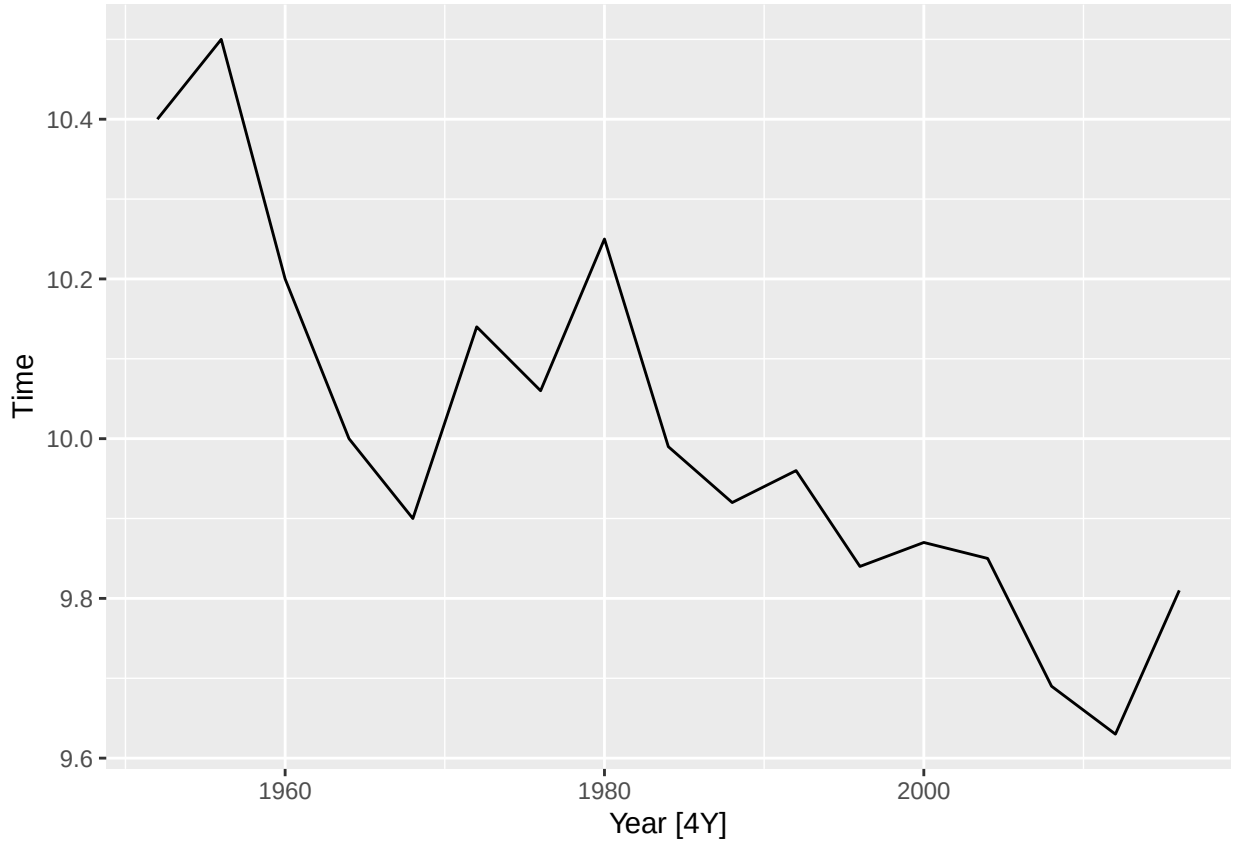
```
## # A tsibble: 312 x 4 [4Y]
## # Key:       Length, Sex [14]
##     Year Length Sex    Time
##    <int>  <int> <chr> <dbl>
##  1  1896    100 men    12
##  2  1900    100 men    11
##  3  1904    100 men    11
##  4  1908    100 men    10.8
##  5  1912    100 men    10.8
##  6  1916    100 men    NA
##  7  1920    100 men    10.8
##  8  1924    100 men    10.6
##  9  1928    100 men    10.8
## 10  1932    100 men    10.3
## # ... with 302 more rows
```

```
olympic_running %>% mutate(Length = as.factor(Length)) %>% ggplot(aes(x=Year, y=Time)) + facet_grid(rows
```

```
## Warning: Removed 31 rows containing missing values (geom_point).
```




```
olympic_running %>% filter(Year >= 1950, Length == 100, Sex== "men") %>% autoplot(Time)
```




```
fit_olympic_running <- olympic_running %>% filter(Year >= 1950, Length == 100, Sex== "men") %>% select('

report(fit_olympic_running)
```

```
## Series: Time
## Model: TSLM
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.25912 -0.04169 -0.01059  0.04757  0.22199
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 29.658235   3.151446   9.411 1.10e-07 ***
## Year        -0.009908   0.001588  -6.238 1.59e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1283 on 15 degrees of freedom
## Multiple R-squared: 0.7218,  Adjusted R-squared: 0.7032
## F-statistic: 38.91 on 1 and 15 DF, p-value: 1.5889e-05
```