Data 624 Homework 5

Steven Gonzalez

3/9/2025

Load Packages

```
library(fpp3)
library(seasonal)
library(USgas)
```

Exercise 1

Consider the the number of pigs slaughtered in Victoria, available in the `aus_livestock` dataset.

Use the `ETS()` function to estimate the equivalent model for simple exponential smoothing. Find the optimal values of alpha and lo, and generate forecasts for the next four months.

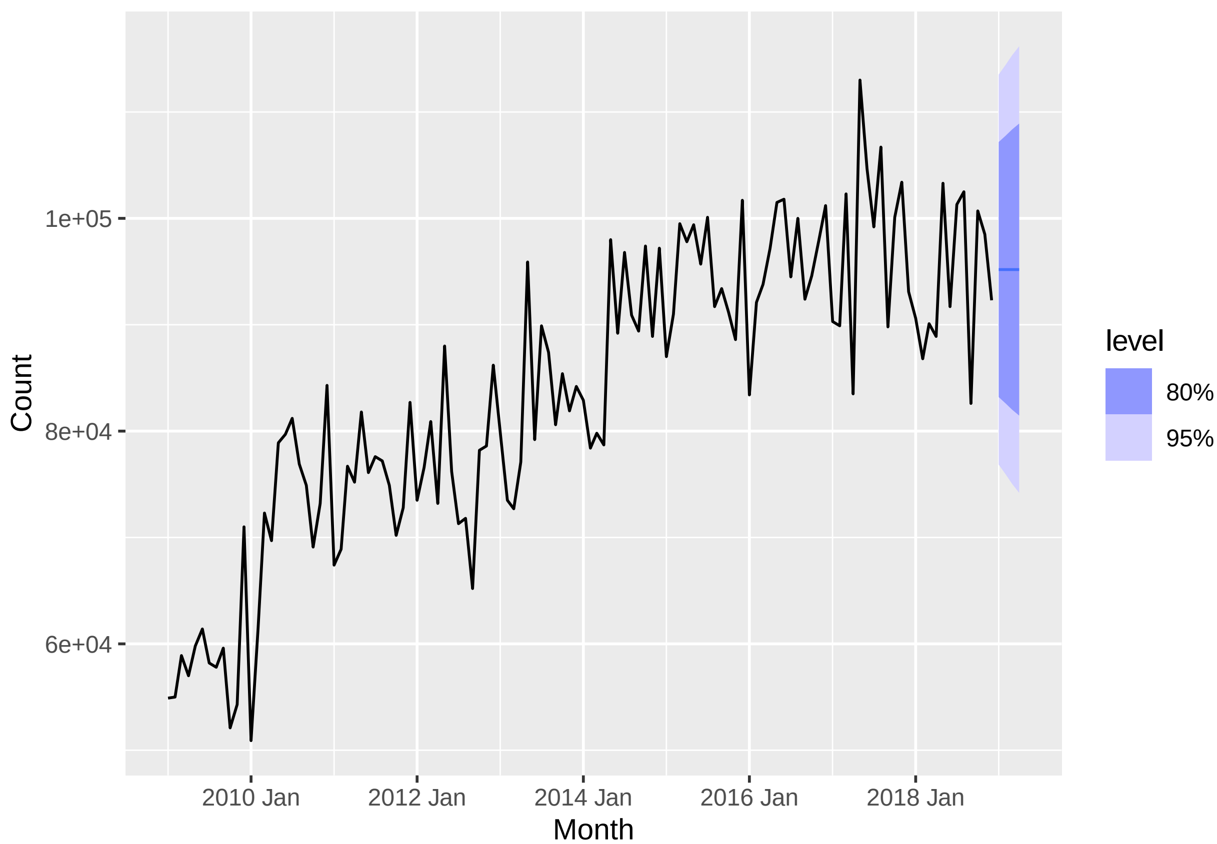
```
vic_pigs <- aus_livestock %>%
  filter(!is.na(Count)) %>%
  filter(State == 'Victoria') %>%
  filter(Animal == 'Pigs')

pfit <- vic_pigs %>%
  model(SES = ETS(Count ~ error('A') + trend('N') + season('N')))

report(pfit)
```

```
## Series: Count
## Model: ETS(A,N,N)
##   Smoothing parameters:
##     alpha = 0.3221247
##
##   Initial states:
##      l[0]
##  100646.6
##
##   sigma^2:  87480760
##
##      AIC     AICc      BIC
## 13737.10 13737.14 13750.07
```

```r
pfit %>%
  forecast(h = 4)
```

```
## # A fable: 4 x 6 [1M]
## # Key:     Animal, State, .model [1]
##   Animal State    .model    Month
##   <fct>  <fct>    <chr>     <mth>
## 1 Pigs   Victoria SES    2019 Jan
## 2 Pigs   Victoria SES    2019 Feb
## 3 Pigs   Victoria SES    2019 Mar
## 4 Pigs   Victoria SES    2019 Apr
## # i 2 more variables: Count <dist>, .mean <dbl>
```

```r
pfit %>%
  forecast(h = 4) %>%
  autoplot(vic_pigs %>%
             filter(year(Month) >= 2009))
```



Compute a 95% prediction interval for the first forecast using y ± 1.96s where s is the standard deviation of the residuals. Compare your interval with the interval produced by R.

```r
pfc <- pfit %>%
  forecast(h = 4)

pfc_s <- sd(augment(pfit)$.resid)
```

```
p_lower_limit <- pfc$.mean[1] - (pfc_s*1.96)
p_upper_limit <- pfc$.mean[1] + (pfc_s*1.96)

cat(p_lower_limit, p_upper_limit)
```

```
## 76871.01 113502.1
```

```
pfc_hilo <- pfc %>%
  hilo()

pfc_hilo$`95%`[1]
```

```
## <hilo[1]>
## [1] [76854.79, 113518.3]95
```

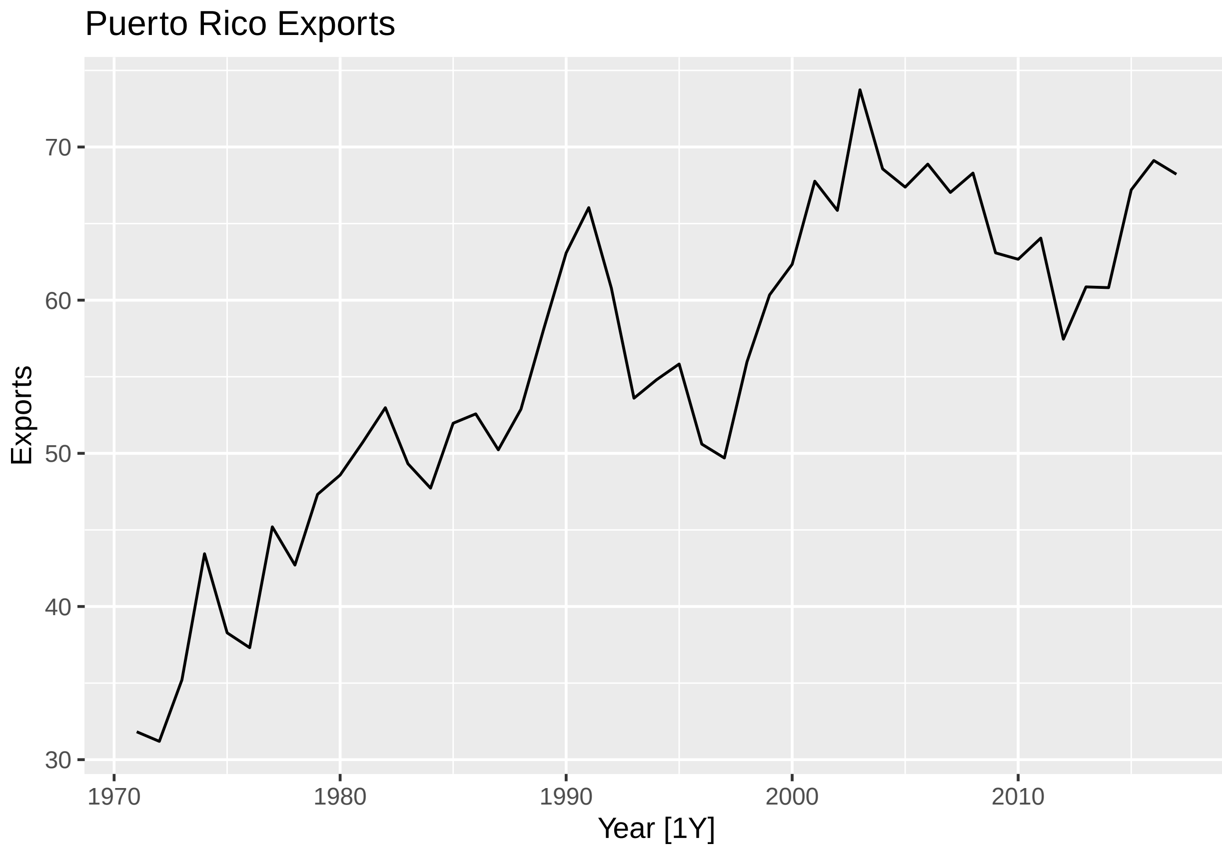
Exercise 5

Data set `global_economy` contains the annual Exports from many countries. Select one country to analyse. Plot the Exports series and discuss the main features of the data.

```
global_economy %>%
  as_tibble() %>%
  select(Country) %>%
  unique()
```

```
## # A tibble: 263 x 1
##    Country
##    <fct>
##  1 Afghanistan
##  2 Albania
##  3 Algeria
##  4 American Samoa
##  5 Andorra
##  6 Angola
##  7 Antigua and Barbuda
##  8 Arab World
##  9 Argentina
## 10 Armenia
## # i 253 more rows
```

```
pr_exp <- global_economy %>%
  filter(!is.na(Exports)) %>%
  filter(Country == 'Puerto Rico')

pr_exp %>%
  autoplot(Exports) +
  labs(title = 'Puerto Rico Exports')
```

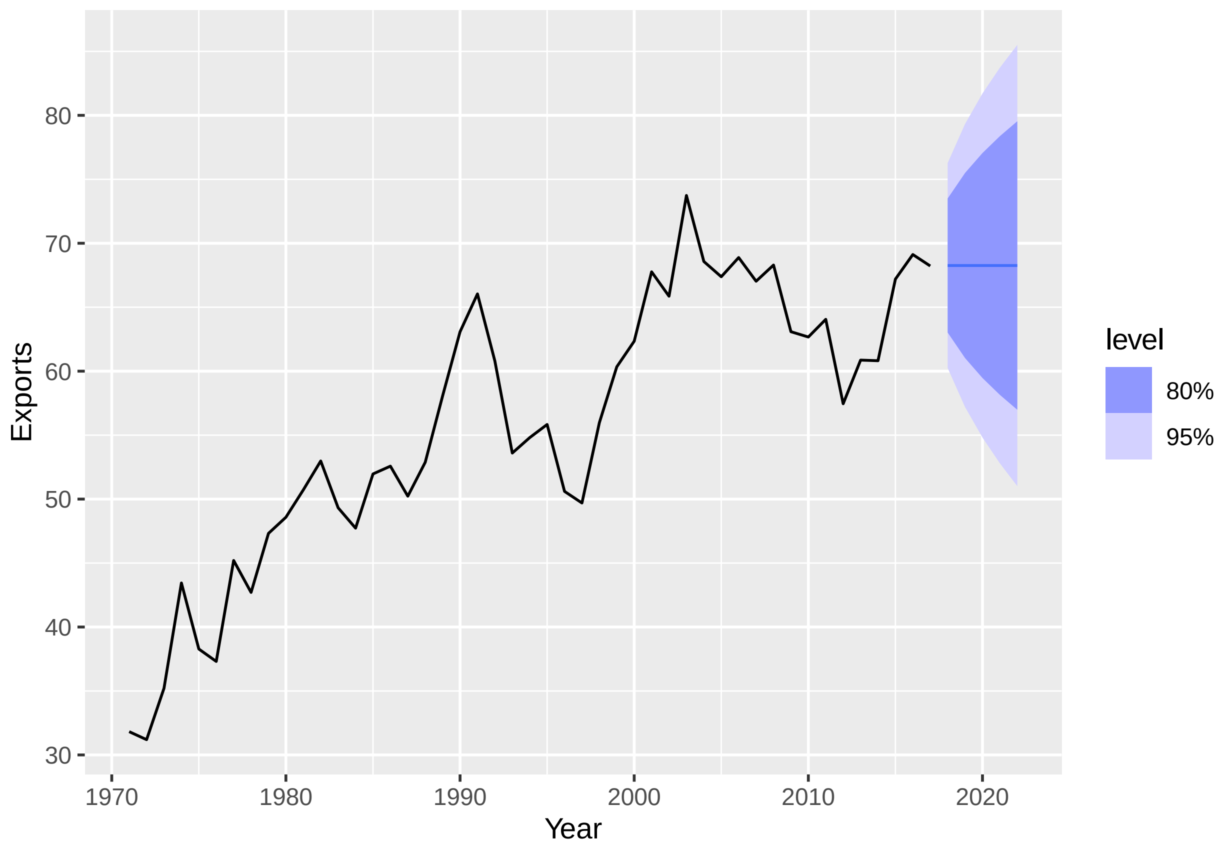


Use an ETS(A,N,N) model to forecast the series, and plot the forecasts.

```
pr_fit <- pr_exp %>%
  model(SES = ETS(Exports ~ error('A') + trend('N') + season('N')))

report(pr_fit)
```

```
## Series: Exports
## Model: ETS(A,N,N)
##   Smoothing parameters:
##     alpha = 0.9535335
##
##   Initial states:
##      l[0]
##  31.79708
##
##   sigma^2:  16.7054
##
##      AIC     AICc      BIC
## 317.2526 317.8108 322.8031
```

```
pr_fit %>%
  forecast(h = 5) %>%
  autoplot(pr_exp)
```



Compute the RMSE values for the training data.

```
pr_fit %>%
  accuracy()
```

```
## # A tibble: 1 x 11
##   Country     .model .type       ME  RMSE   MAE   MPE  MAPE  MASE RMSSE    ACF1
##   <fct>       <chr>  <chr>    <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>   <dbl>
## 1 Puerto Rico SES    Training 0.814  4.00  3.28  1.40  6.03 0.981 0.988 -0.0414
```

Compare the results to those from an ETS(A,A,N) model. (Remember that the trended model is using one more parameter than the simpler model.) Discuss the merits of the two forecasting methods for this data set.

```
pr_fit1 <- pr_exp %>%
  model(Holt = ETS(Exports ~ error('A') + trend('A') + season('N')))

report(pr_fit1)
```

```
## Series: Exports
## Model: ETS(A,A,N)
##   Smoothing parameters:
##     alpha = 0.8870112
##     beta  = 0.0001000318
##
```

```
##   Initial states:
##      l[0]      b[0]
##  29.19841 0.8219231
##
##   sigma^2:  16.764
##
##      AIC     AICc      BIC
## 319.2805 320.7439 328.5312
```

```r
pr_fit1 %>%
  accuracy()
```

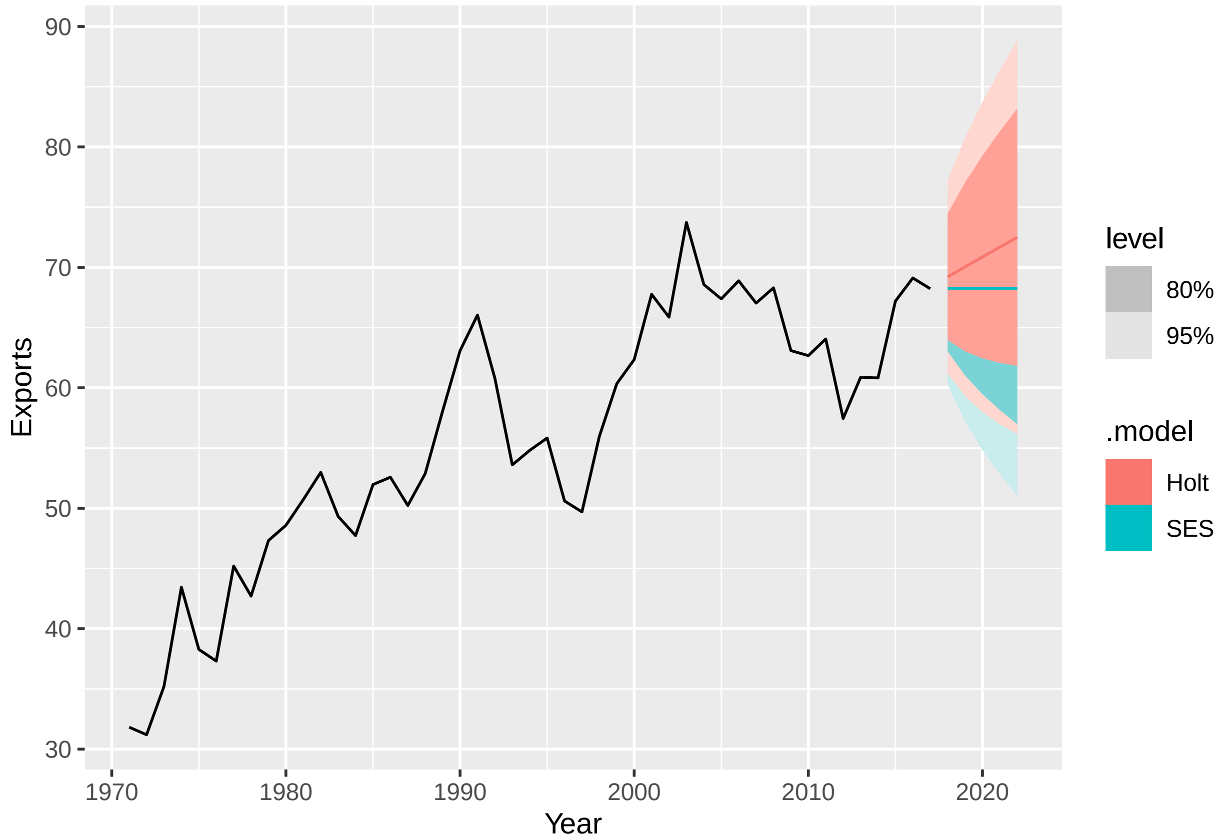
```
## # A tibble: 1 x 11
##   Country     .model .type       ME  RMSE   MAE     MPE  MAPE  MASE RMSSE   ACF1
##   <fct>       <chr>  <chr>    <dbl> <dbl> <dbl>   <dbl> <dbl> <dbl> <dbl>  <dbl>
## 1 Puerto Rico Holt   Traini~ 0.0127  3.92  3.22 -0.0265  6.01 0.962 0.968 0.0144
```

Compare the forecasts from both methods. Which do you think is best?

```r
pr_exp %>%
  model(SES = ETS(Exports ~ error('A') + trend('N') + season('N')),
        Holt = ETS(Exports ~ error('A') + trend('A') + season('N'))) %>%
  accuracy()
```

```
## # A tibble: 2 x 11
##   Country     .model .type      ME  RMSE   MAE     MPE  MAPE  MASE RMSSE    ACF1
##   <fct>       <chr>  <chr>   <dbl> <dbl> <dbl>   <dbl> <dbl> <dbl> <dbl>   <dbl>
## 1 Puerto Rico SES    Train~ 0.814   4.00  3.28  1.40    6.03 0.981 0.988 -0.0414
## 2 Puerto Rico Holt   Train~ 0.0127  3.92  3.22 -0.0265  6.01 0.962 0.968  0.0144
```

```r
pr_exp %>%
  model(SES = ETS(Exports ~ error('A') + trend('N') + season('N')),
        Holt = ETS(Exports ~ error('A') + trend('A') + season('N'))) %>%
  forecast(h = 5) %>%
  autoplot(pr_exp)
```



Calculate a 95% prediction interval for the first forecast for each model, using the RMSE values and assuming normal errors. Compare your intervals with those produced using R.

```
pr_fc <- pr_fit %>%
  forecast(h = 5)

pr_fc_s <- sd(augment(pr_fit)$.resid)

pr_lower_limit <- pr_fc$.mean[1] - (pr_fc_s*1.96)
pr_upper_limit <- pr_fc$.mean[1] + (pr_fc_s*1.96)

cat(pr_lower_limit, pr_upper_limit)
```

```
## 60.50336 76.01878
```

```
pr_fc_hilo <- pr_fc %>%
  hilo()

pr_fc_hilo$`95%`[1]
```

```
## <hilo[1]>
## [1] [60.25025, 76.27189]95
```

Above is the comparison using the SES model.

```
pr_fc1 <- pr_fit1 %>%
  forecast(h = 5)

pr_fc_s1 <- sd(augment(pr_fit1)$.resid)

pr1_lower_limit <- pr_fc1$.mean[1] - (pr_fc_s1*1.96)
pr1_upper_limit <- pr_fc1$.mean[1] + (pr_fc_s1*1.96)

cat(pr1_lower_limit, pr1_upper_limit)
```

```
## 61.45942 76.97715
```

```
pr_fc_hilo1 <- pr_fc1 %>%
  hilo()

pr_fc_hilo1$`95%`[1]
```

```
## <hilo[1]>
## [1] [61.19343, 77.24314]95
```

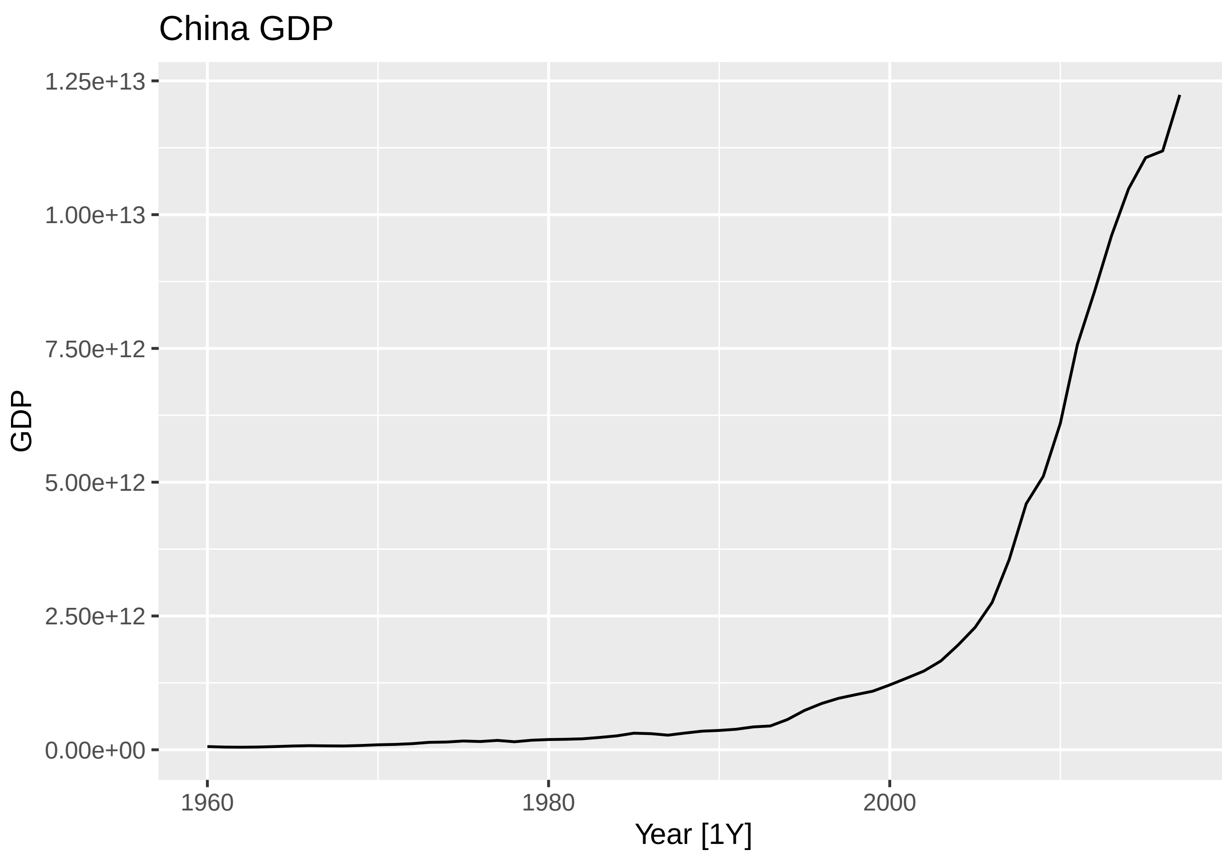
Above is the comparison using the Holt model.

Exercise 6

Forecast the Chinese GDP from the `global_economy` data set using an ETS model. Experiment with the various options in the `ETS()` function to see how much the forecasts change with damped trend, or with a Box-Cox transformation. Try to develop an intuition of what each is doing to the forecasts.

[Hint: use a relatively large value of `h` when forecasting, so you can clearly see the differences between the various options when plotting the forecasts.]

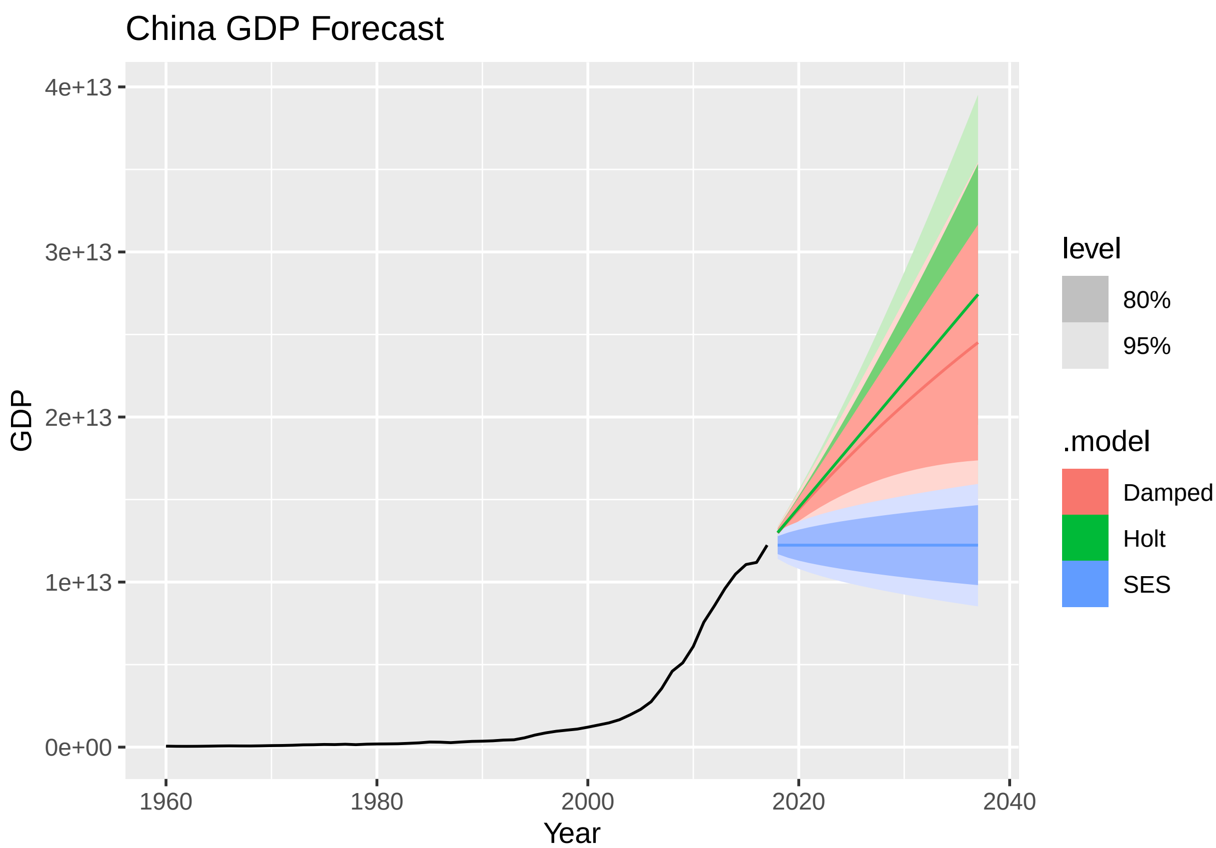
```
china_GDP <- global_economy %>%
  filter(!is.na(Population)) %>%
  filter(Country == 'China')

china_GDP %>%
  autoplot(GDP) +
  labs(title = 'China GDP')
```




```
cfit <- china_GDP %>%
  model(SES = ETS(GDP ~ error('A') + trend('N') + season('N')),
        Holt = ETS(GDP ~ error('A') + trend('A') + season('N')),
        Damped = ETS(GDP ~ error('A') + trend('Ad') + season('N')))

cfit %>%
  accuracy()
```

```
## # A tibble: 3 x 11
##   Country .model .type        ME    RMSE     MAE   MPE  MAPE  MASE RMSSE     ACF1
##   <fct>   <chr>  <chr>     <dbl>   <dbl>   <dbl> <dbl> <dbl> <dbl> <dbl>    <dbl>
## 1 China   SES    Train~ 2.10e11 4.16e11 2.13e11  8.14 11.0  0.983 0.991  0.789
## 2 China   Holt   Train~ 2.36e10 1.90e11 9.59e10  1.41  7.62 0.442 0.453  0.00905
## 3 China   Damped Train~ 2.95e10 1.90e11 9.49e10  1.62  7.62 0.438 0.454 -0.00187
```

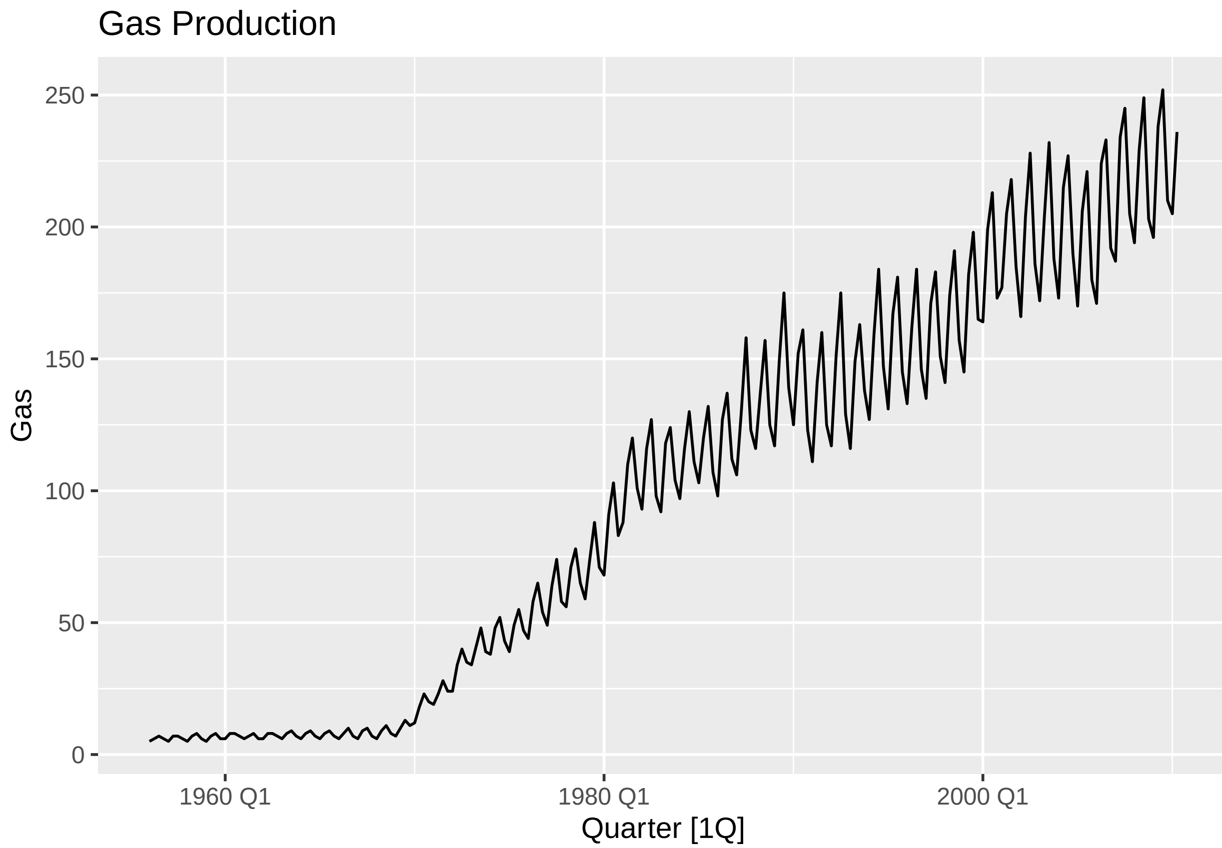
```
cfit %>%
  forecast(h = 20) %>%
  autoplot(china_GDP) +
  labs(title = 'China GDP Forecast')
```



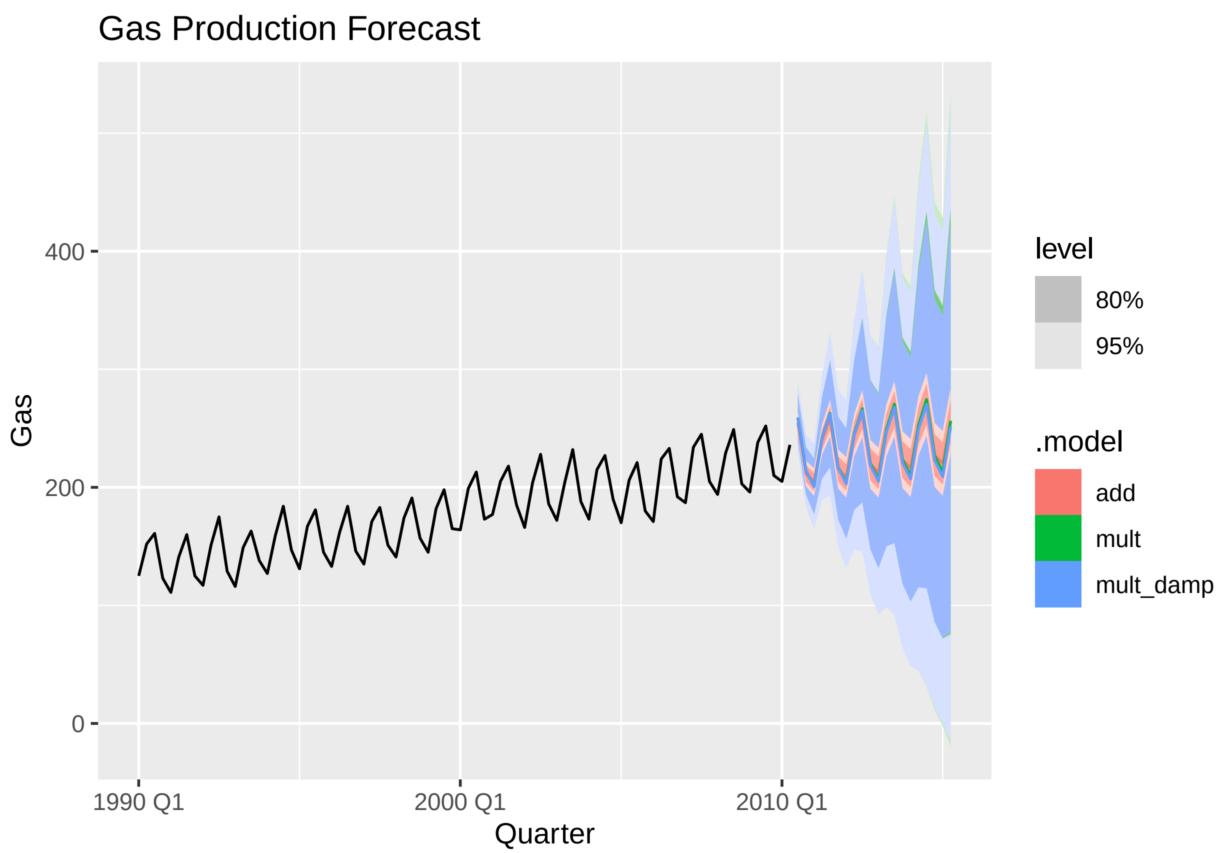
Exercise 7

Find an ETS model for the Gas data from `aus_production` and forecast the next few years. Why is multiplicative seasonality necessary here? Experiment with making the trend damped. Does it improve the forecasts?

```
gas <- aus_production %>%
  filter(!is.na(Gas)) %>%
  select(Gas)

autoplot(gas, Gas) +
  labs(title = 'Gas Production')
```




```
gfit <- gas %>%
  model(add = ETS(Gas ~ error('A') + trend('A') + season('A')),
        mult = ETS(Gas ~ error('M') + trend('A') + season('M')),
        mult_damp = ETS(Gas ~ error('M') + trend('Ad') + season('M')))

gfit %>%
  forecast(h = 20) %>%
  autoplot(gas %>%
             filter(year(Quarter) >= 1990)) +
  labs(title = 'Gas Production Forecast')
```



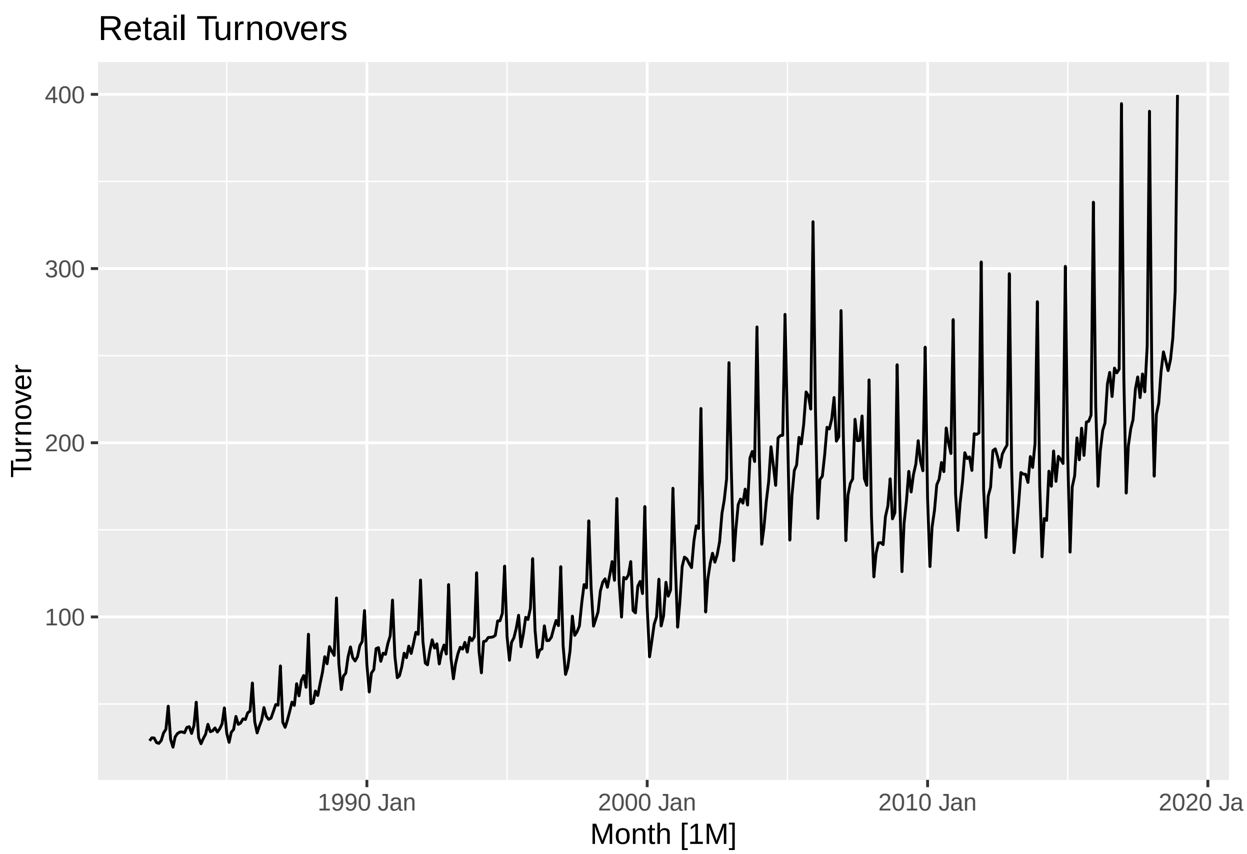

```r
gfit %>%
  accuracy()
```

```
## # A tibble: 3 x 10
##   .model     .type          ME  RMSE   MAE    MPE  MAPE  MASE RMSSE    ACF1
##   <chr>      <chr>       <dbl> <dbl> <dbl>  <dbl> <dbl> <dbl> <dbl>   <dbl>
## 1 add        Training  0.00525  4.76  3.35 -4.69  10.9  0.600 0.628  0.0772
## 2 mult       Training -0.115    4.60  3.02  0.199  4.08 0.542 0.606 -0.0131
## 3 mult_damp  Training -0.00439  4.59  3.03  0.326  4.10 0.544 0.606 -0.0217
```

Exercise 8

Recall your retail time series data (from Exercise 7 in Section 2.10).

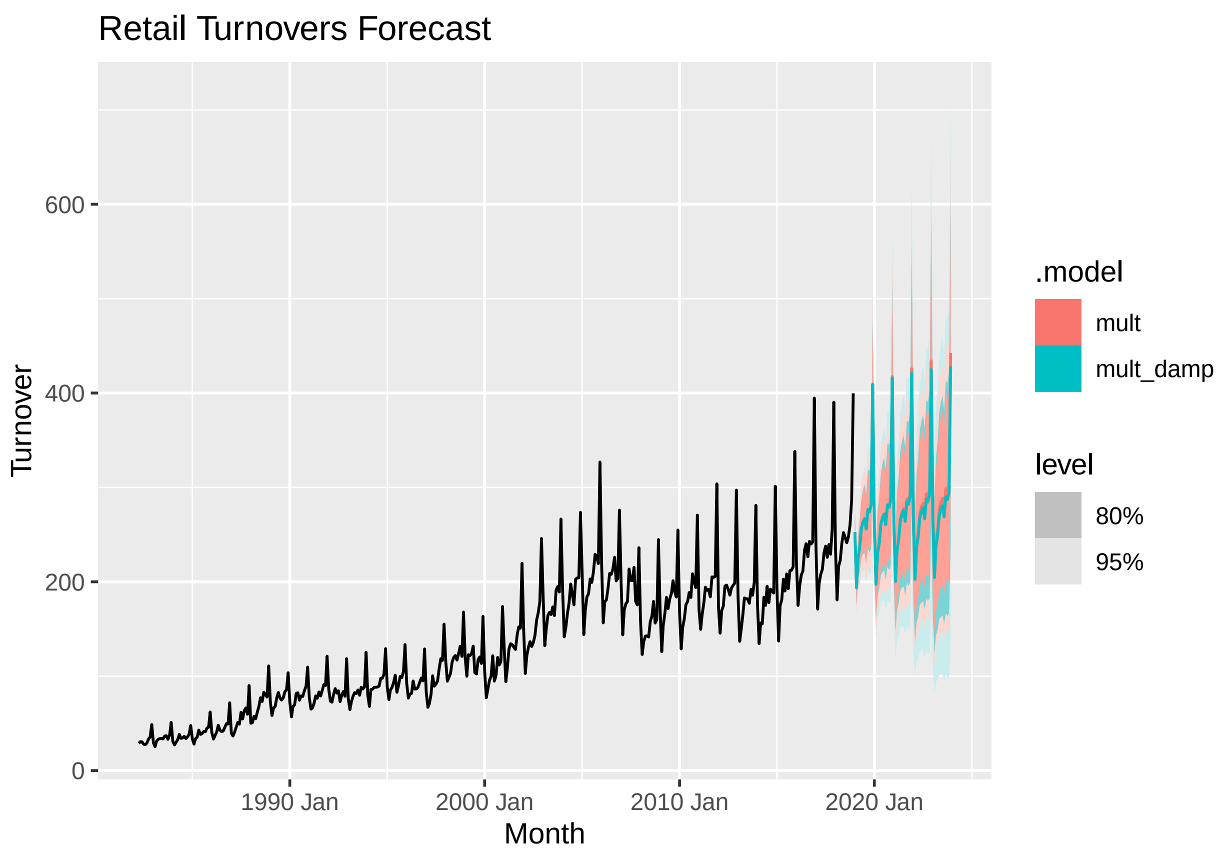
```r
set.seed(1)
myseries <- aus_retail %>%
  filter(`Series ID` == sample(aus_retail$`Series ID`,1))

autoplot(myseries) +
  labs(title = 'Retail Turnovers')
```



Why is multiplicative seasonality necessary for this series?

Apply Holt-Winters' multiplicative method to the data. Experiment with making the trend damped.

```
mfit <- myseries %>%
  model(mult = ETS(Turnover ~ error('M') + trend('A') + season('M')),
        mult_damp = ETS(Turnover ~ error('M') + trend('Ad') + season('M')))

mfit %>%
  forecast(h = 60) %>%
  autoplot(myseries) +
  labs(title = 'Retail Turnovers Forecast')
```



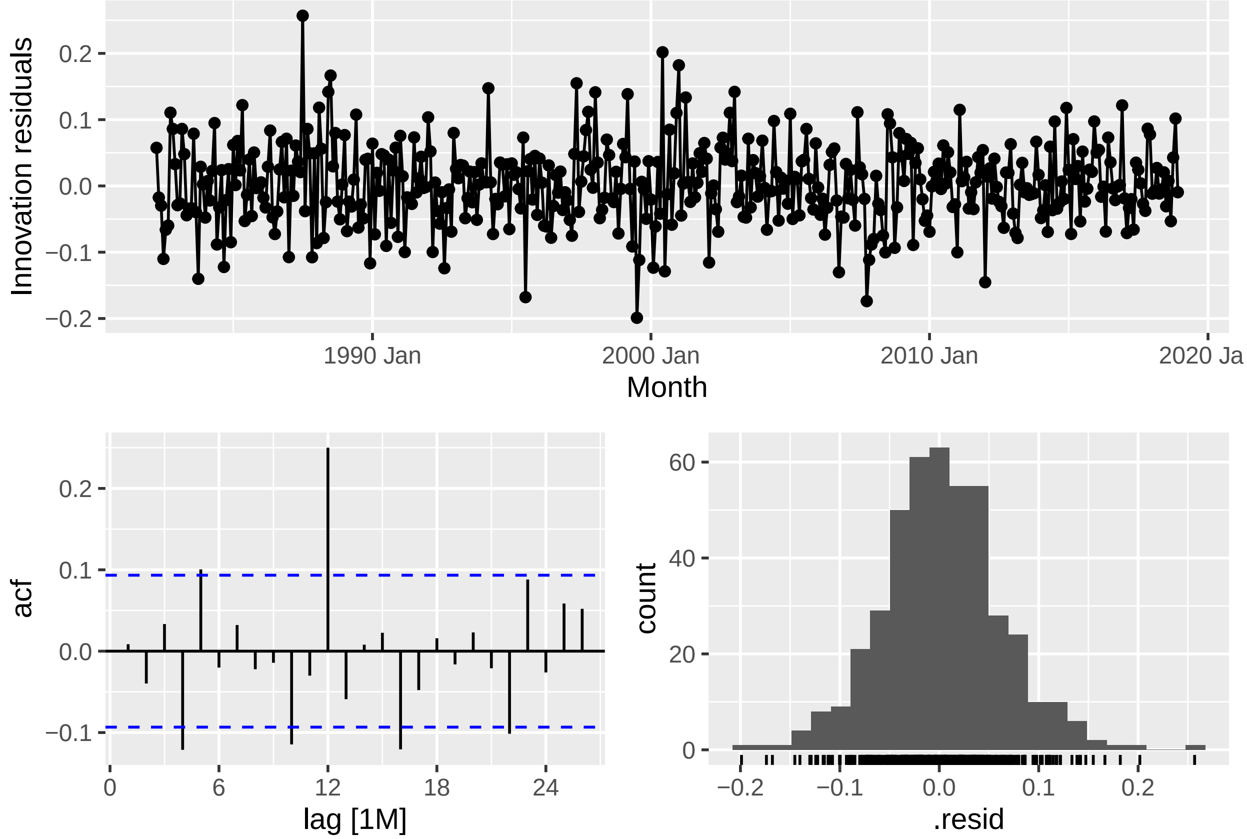
Compare the RMSE of the one-step forecasts from the two methods. Which do you prefer?

```
mfit %>%
  accuracy()
```

```
## # A tibble: 2 x 12
##   State Industry .model .type    ME  RMSE   MAE     MPE  MAPE  MASE RMSSE   ACF1
##   <chr> <chr>    <chr>  <chr> <dbl> <dbl> <dbl>   <dbl> <dbl> <dbl> <dbl>  <dbl>
## 1 Quee~ Clothin~ mult   Trai~ 0.415  8.53  5.95 -0.0595  4.65 0.483 0.511 0.0814
## 2 Quee~ Clothin~ mult_~ Trai~ 0.689  8.55  5.99  0.174   4.66 0.485 0.512 0.0588
```

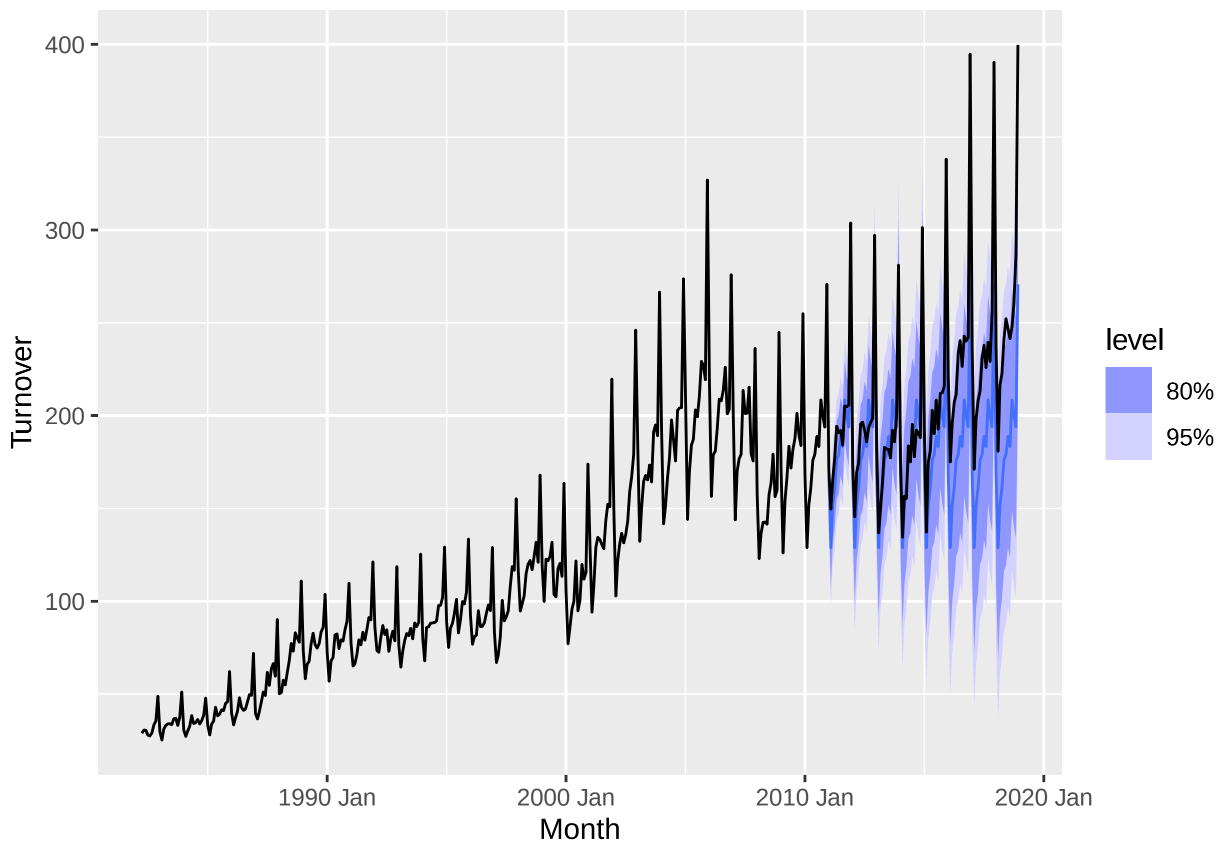
Check that the residuals from the best method look like white noise.

```
mfit1 <- myseries %>%
  model(mult = ETS(Turnover ~ error('M') + trend('A') + season('M')))

mfit1 %>% gg_tsresiduals()
```



Now find the test set RMSE, while training the model to the end of 2010. Can you beat the seasonal naïve approach from Exercise 7 in Section 5.11?

```
myseries_train <- myseries %>%
  filter(year(Month) < 2011)

mfit_sn <- myseries_train %>%
  model(SNAIVE(Turnover))

fc <- mfit_sn %>%
  forecast(new_data = anti_join(myseries, myseries_train))
fc %>%
  autoplot(myseries)
```



Exercise 9

For the same retail data, try an STL decomposition applied to the Box-Cox transformed series, followed by ETS on the seasonally adjusted data. How does that compare with your best previous forecasts on the test set?