Retail Project

Marcie Joylynn Luke

12/05/2022

```
library(fpp3)
```

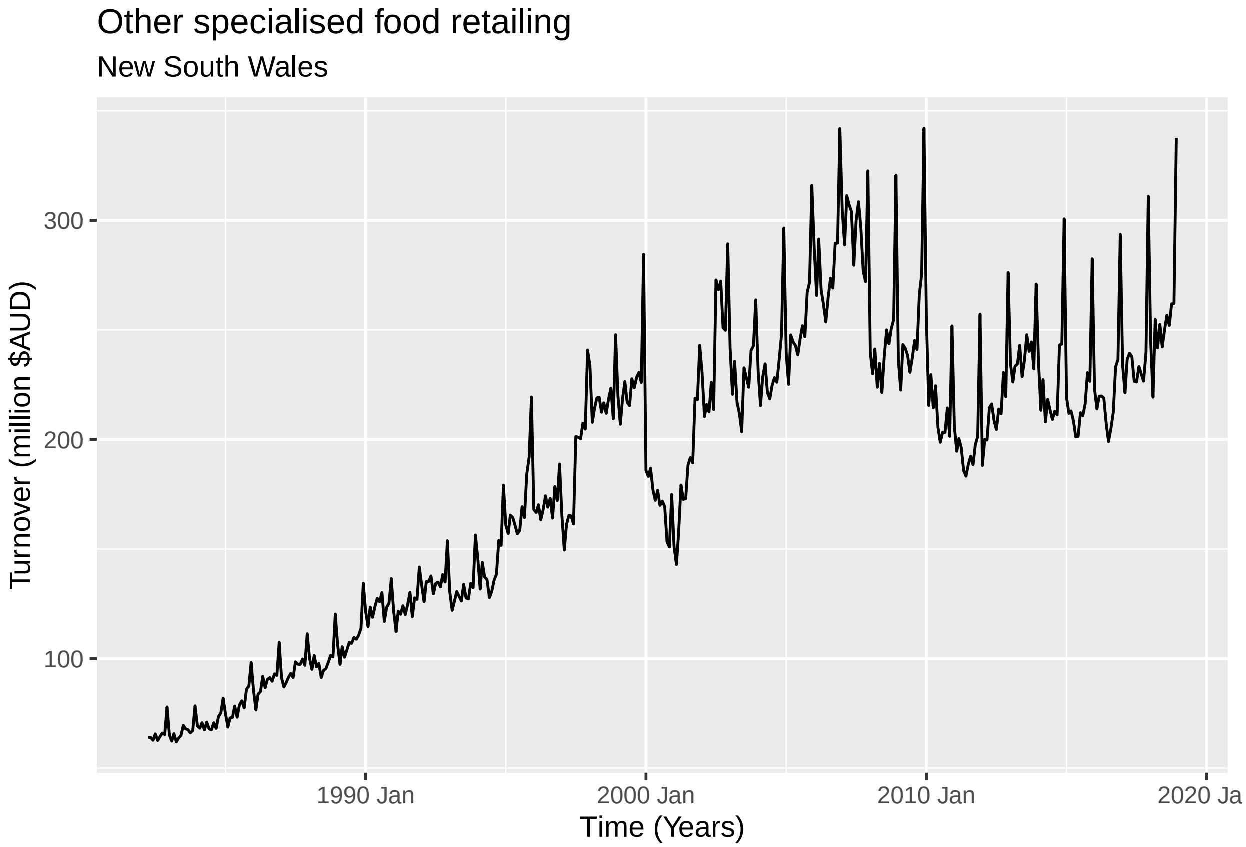
```
## -- Attaching packages -------------------------------------------- fpp3 0.4.0 --

## v tibble      3.1.6     v tsibble     1.1.1
## v dplyr       1.0.8     v tsibbledata 0.4.0
## v tidyr       1.2.0     v feasts      0.2.2
## v lubridate   1.8.0     v fable       0.3.1
## v ggplot2     3.3.5

## -- Conflicts ------------------------------------------------- fpp3_conflicts --
## x lubridate::date()    masks base::date()
## x dplyr::filter()      masks stats::filter()
## x tsibble::intersect() masks base::intersect()
## x tsibble::interval()  masks lubridate::interval()
## x dplyr::lag()         masks stats::lag()
## x tsibble::setdiff()   masks base::setdiff()
## x tsibble::union()     masks base::union()
```

Statistical Features of the Data

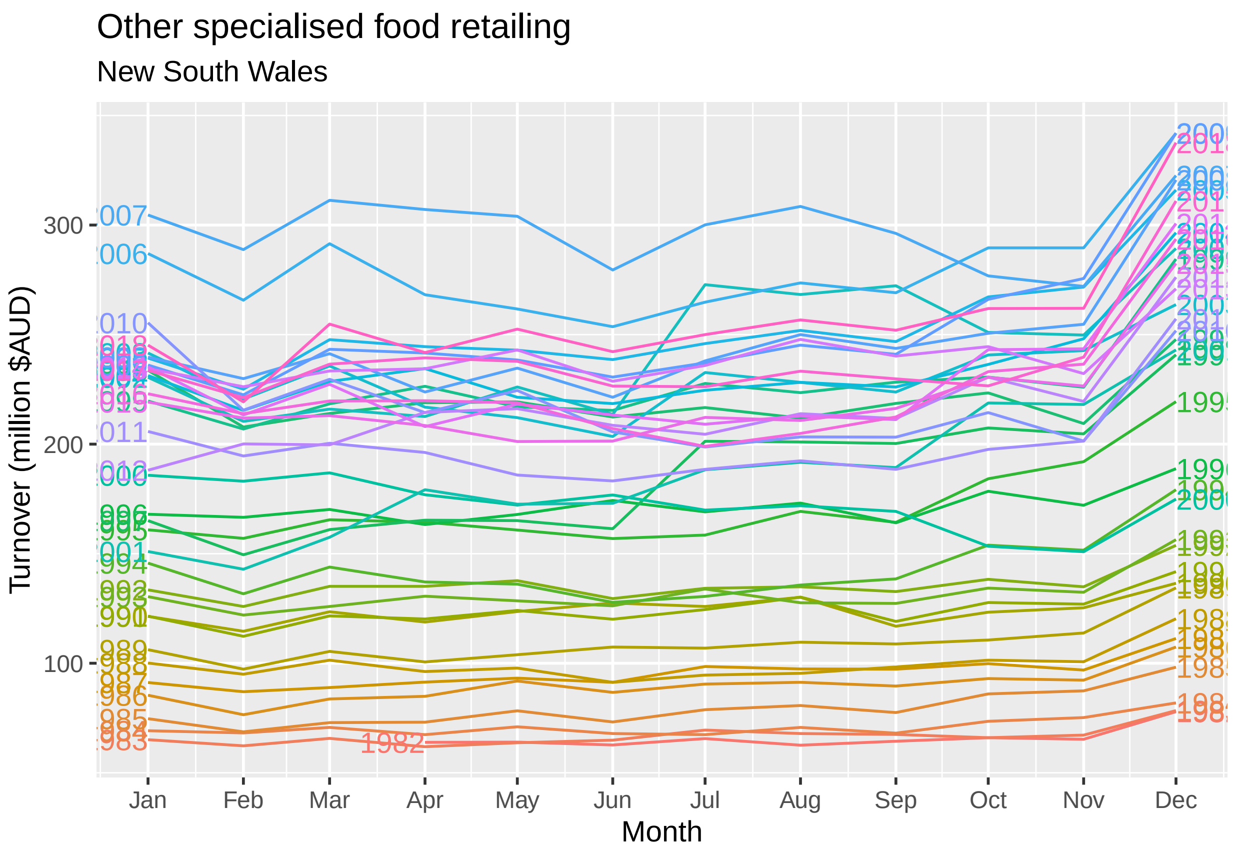
Overtime, we could see there is an upward trend, although there has been a lot of volatility in the data. Turnover tend to fall in February and increase in December which is its highest turnover for the year. The magnitude of the changes in turnover seems to be more volatile for the more recent years.

This seasonality pattern observed previously from seasonality graph also supported by graph of the subseries, where it could be seen that average of the turnover for December across the years is the highest compared to other months, as well as February having the lowest average turnover across the years. Turnover across years have been increasing for each month, on average. It is also observed that 2005 seems to have a very significant higher turnover for each months compared to other years.

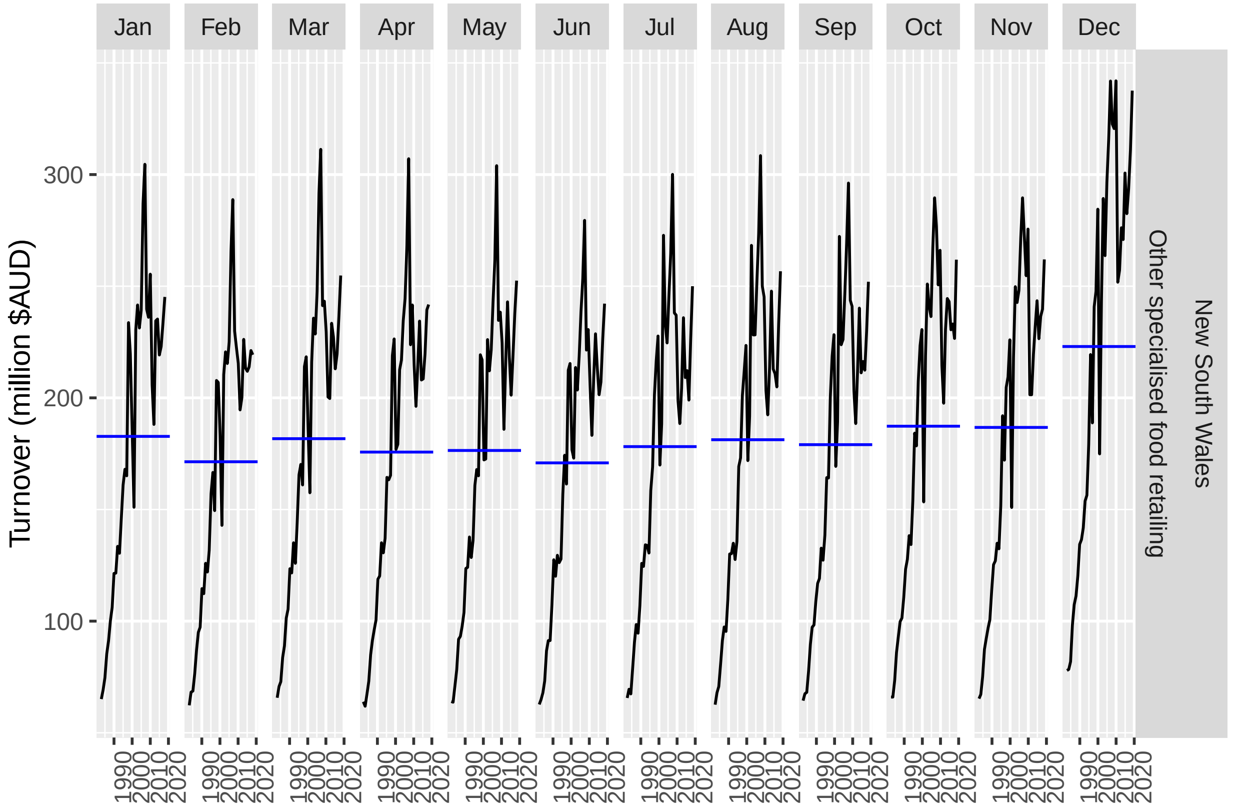
```
myseries %>%
  autoplot(Turnover) +
  labs(y = "Turnover (million $AUD)", x = "Time (Years)",
       title = myseries$Industry[1],
       subtitle = myseries$State[1])
```



```
myseries %>%
  gg_season(Turnover, labels = "both") +
  labs(y = "Turnover (million $AUD)",
       title = myseries$Industry[1],
       subtitle = myseries$State[1])
```



```
myseries %>%
  gg_subseries(Turnover) +
  labs(y = "Turnover (million $AUD)", x="")
```



A. ETS model

1. Fit the model

The best ETS model is model with multiplicative error, additional damped trend, and multiplicative seasonality. This could be obtained automatically without specifying specification of the ets model when training the data.

We could also compared this model with other ETS model to find out which one has the lowest AIC. In this case, M,N,M(multiplicative error,no trend, and multiplicative season) and M,A,M(multiplicative error,additive trend, and multiplicative season) model are chosen as comparison to M,Ad,M model.

It could be shown that M,Ad, M model is the best model for the retail data as it has lowest AIC.

```
myseries_tr <- myseries %>% head(417)
best_ets <- myseries_tr %>% model(best=ETS(Turnover))
try_ets <- myseries_tr %>% model(best=ETS(Turnover),mnm=ETS(Turnover~error("M")+trend("N")+season("M"))

glance(try_ets) %>% select(State:AICc) %>% arrange(AIC)
```

```
## # A tibble: 3 x 7
##   State           Industry                          .model  sigma2 log_lik   AIC  AICc
##   <chr>           <chr>                             <chr>    <dbl>   <dbl> <dbl> <dbl>
## 1 New South Wales Other specialised food ret~ best   0.00211  -2091. 4219. 4220.
## 2 New South Wales Other specialised food ret~ mam    0.00227  -2108. 4249. 4251.
## 3 New South Wales Other specialised food ret~ mnm    0.00282  -2152. 4335. 4336.
```

```
report(best_ets)
```

```
## Series: Turnover
## Model: ETS(M,Ad,M)
##   Smoothing parameters:
##     alpha = 0.7799953
##     beta  = 0.002644142
##     gamma = 0.0001021286
##     phi   = 0.9742997
##
##   Initial states:
##      l[0]      b[0]      s[0]     s[-1]    s[-2]    s[-3]    s[-4]    s[-5]
##  64.26376 0.2187576 0.9858752 0.9376403 1.004066 1.188357 1.014641 1.025215
##      s[-6]     s[-7]     s[-8]    s[-9]    s[-10]    s[-11]
##  0.9766161 0.9934338 0.9856194 0.944484 0.9736245 0.9704284
##
##   sigma^2:  0.0021
##
##      AIC     AICc      BIC
## 4218.578 4220.297 4291.174
```

```
tidy(best_ets)
```

```
## # A tibble: 18 x 5
##    State           Industry                         .model term    estimate
##    <chr>           <chr>                            <chr>  <chr>      <dbl>
##  1 New South Wales Other specialised food retailing best   alpha   0.780
##  2 New South Wales Other specialised food retailing best   beta    0.00264
##  3 New South Wales Other specialised food retailing best   gamma   0.000102
##  4 New South Wales Other specialised food retailing best   phi     0.974
##  5 New South Wales Other specialised food retailing best   l[0]   64.3
##  6 New South Wales Other specialised food retailing best   b[0]    0.219
##  7 New South Wales Other specialised food retailing best   s[0]    0.986
##  8 New South Wales Other specialised food retailing best   s[-1]   0.938
##  9 New South Wales Other specialised food retailing best   s[-2]   1.00
## 10 New South Wales Other specialised food retailing best   s[-3]   1.19
## 11 New South Wales Other specialised food retailing best   s[-4]   1.01
## 12 New South Wales Other specialised food retailing best   s[-5]   1.03
## 13 New South Wales Other specialised food retailing best   s[-6]   0.977
## 14 New South Wales Other specialised food retailing best   s[-7]   0.993
## 15 New South Wales Other specialised food retailing best   s[-8]   0.986
## 16 New South Wales Other specialised food retailing best   s[-9]   0.944
## 17 New South Wales Other specialised food retailing best   s[-10]  0.974
## 18 New South Wales Other specialised food retailing best   s[-11]  0.970
```

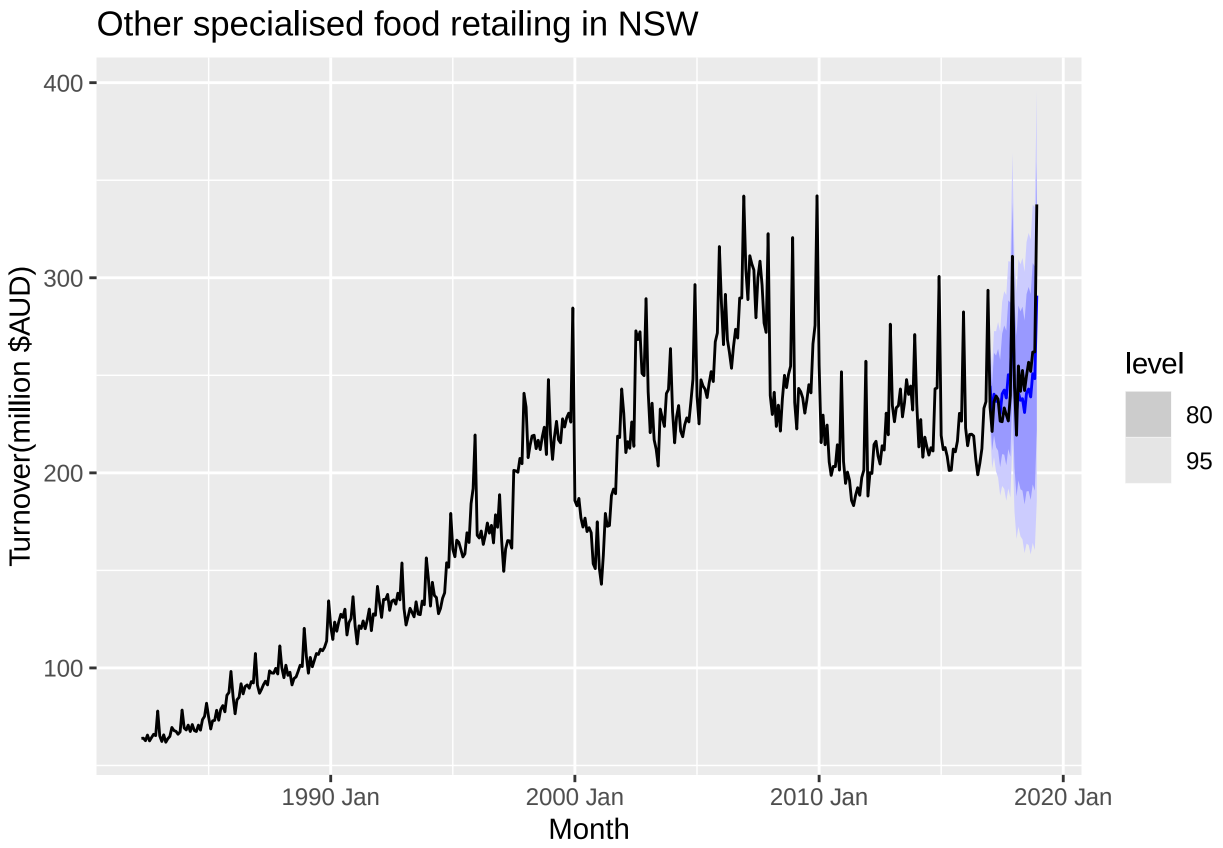
2. Produce Forecast

Forecast for the last 24 months of the original data is produced, as well as its 80% prediction interval. The plot comparing th actual values and the forecasted values is shown below. It could be seen that the model has captured the the information well, as the forecasted value is quite similar to the actual values.

```
test_fc_ets <- best_ets %>% forecast(h="24 months")
interval <- test_fc_ets %>% mutate(interval=hilo(Turnover,0.80)) %>% pull(interval)
test_fc_ets <- test_fc_ets %>% mutate(Interval=interval)
test_fc_ets
```

```
## # A fable: 24 x 7 [1M]
## # Key:     State, Industry, .model [1]
##    State      Industry .model    Month    Turnover .mean                Interval
##    <chr>      <chr>    <chr>     <mth>      <dist> <dbl>                  <hilo>
##  1 New South~ Other s~ best   2017 Jan N(245, 126)  245. [244.5254, 244.7508]0.8
##  2 New South~ Other s~ best   2017 Feb N(229, 178)  229. [228.3803, 228.6478]0.8
##  3 New South~ Other s~ best   2017 Mar N(240, 272)  240. [240.1632, 240.4941]0.8
##  4 New South~ Other s~ best   2017 Apr N(237, 337)  237. [236.4342, 236.8025]0.8
##  5 New South~ Other s~ best   2017 May N(237, 414)  237. [237.2509, 237.6592]0.8
##  6 New South~ Other s~ best   2017 Jun N(230, 461)  230. [230.1834, 230.6139]0.8
##  7 New South~ Other s~ best   2017 Jul N(240, 580)  240. [240.2435, 240.7264]0.8
##  8 New South~ Other s~ best   2017 Aug N(242, 669)  242. [242.1898, 242.7084]0.8
##  9 New South~ Other s~ best   2017 Sep N(238, 724)  238. [238.1250, 238.6645]0.8
## 10 New South~ Other s~ best   2017 Oct N(250, 884)  250. [250.0078, 250.6039]0.8
## # ... with 14 more rows
```

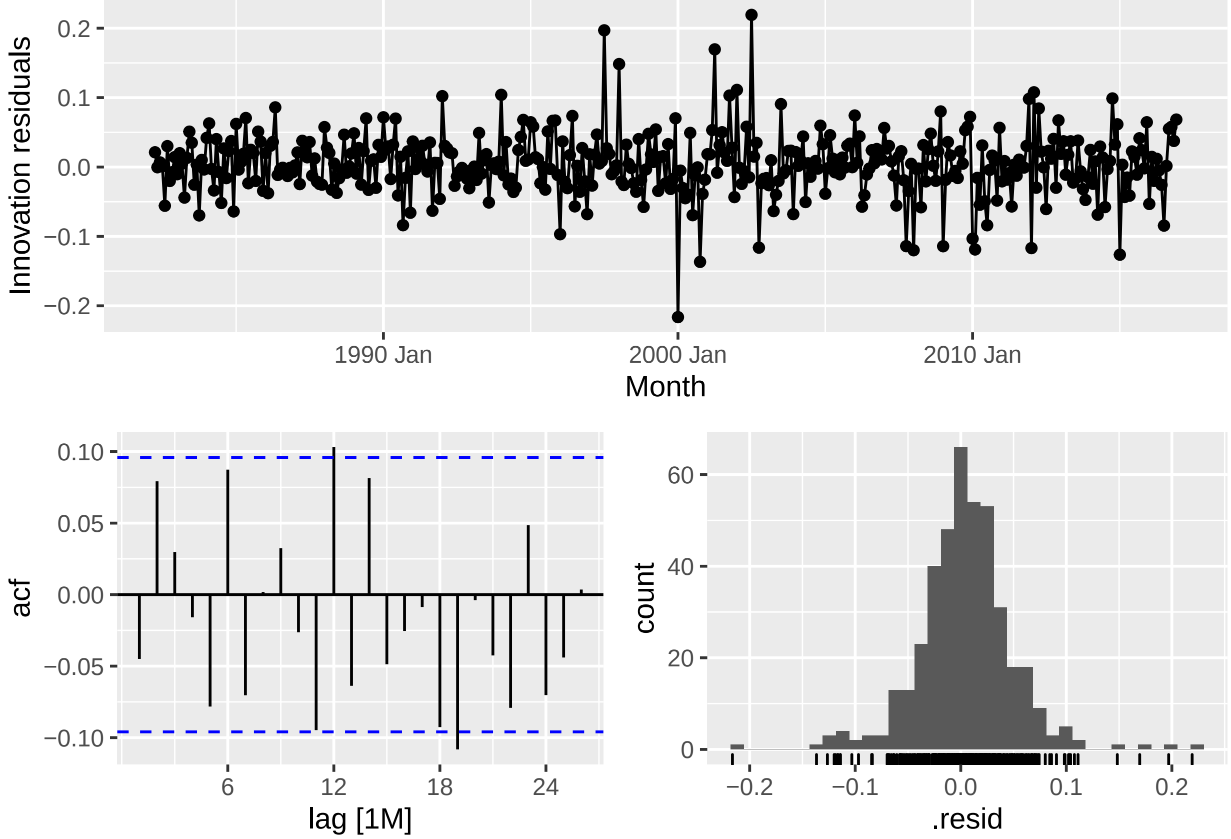
```
best_ets %>% forecast(h="24 months") %>% autoplot(myseries)+labs(title="Other specialised food retailin
```



3. Residual Diagnostic

As shown from the acf of the residual, there are still some correlation in the residual, which means the model has not fully capture the available information. This is also supported by the ljung-box test (pval=6.476627e-05), where we can reject the null that there is no serial correlation in the residual up to 36 lags, at 5% level of significant.

```
best_ets %>% gg_tsresiduals()
```



```
best_ets %>%
  augment() %>%
  features(.innov, ljung_box, dof = 18, lag = 36)
```

```
## # A tibble: 1 x 5
##   State           Industry                              .model lb_stat lb_pvalue
##   <chr>           <chr>                                 <chr>    <dbl>     <dbl>
## 1 New South Wales Other specialised food retailing best      50.4 0.0000648
```

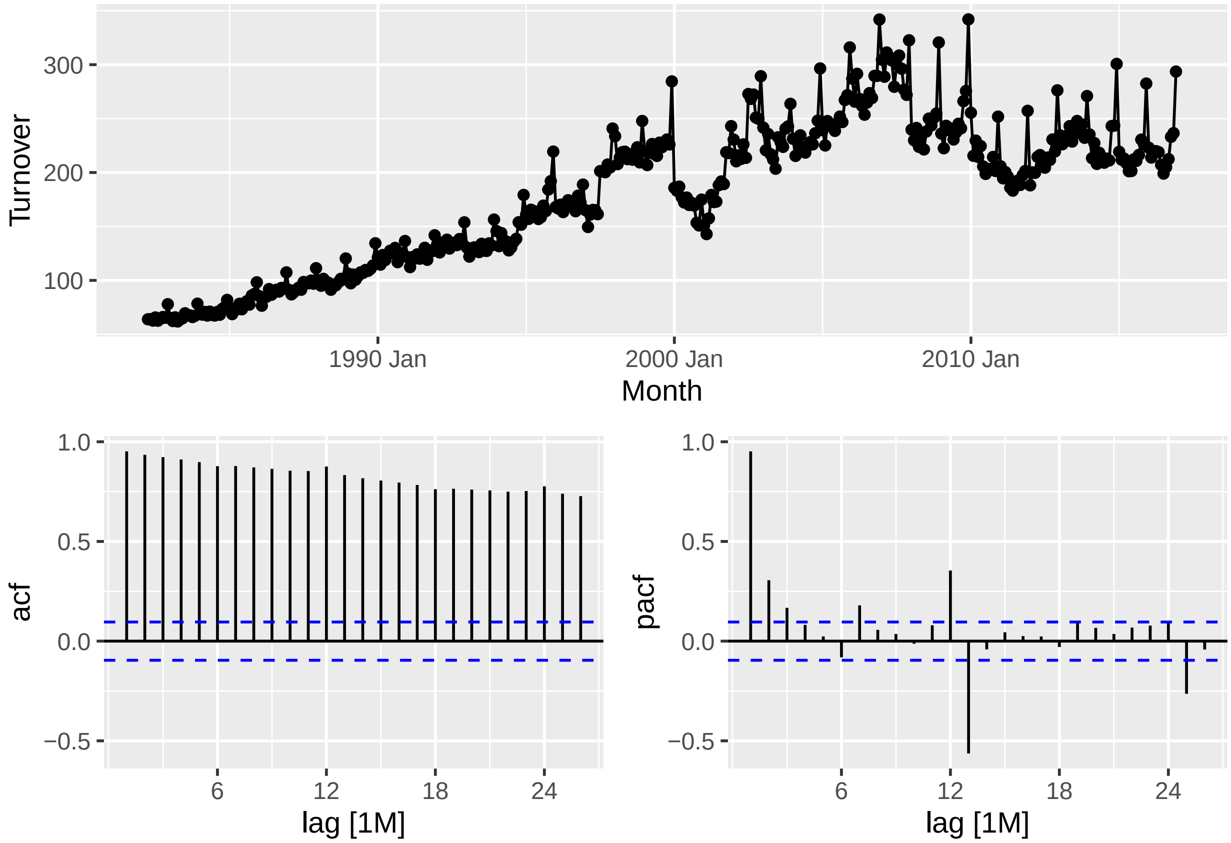
B. ARIMA Model

1. Checking for Stationarity

For ARIMA model, it is important that the data to be fitted is stationary. It could be seen from the graph, the existence of the upward trend and seasonality in the data. From the ACF, it could be seen that it is decaying slowly.Thus, we can conclude that the data is not stationary. Furthermore, unit root test is also performed. As pvalue(0.01)<0.05, we can reject the null that there is no unit root, at 5% level of significance. Thus, we also conclude the data is not stationary.

```
myseries_tr %>% gg_tsdisplay(plot_type="partial")
```

```
## Plot variable not specified, automatically selected ‘y = Turnover‘
```



```
myseries_tr %>% features(Turnover,unitroot_kpss)
```

```
## # A tibble: 1 x 4
##   State           Industry                         kpss_stat kpss_pvalue
##   <chr>           <chr>                                <dbl>       <dbl>
## 1 New South Wales Other specialised food retailing      5.82        0.01
```

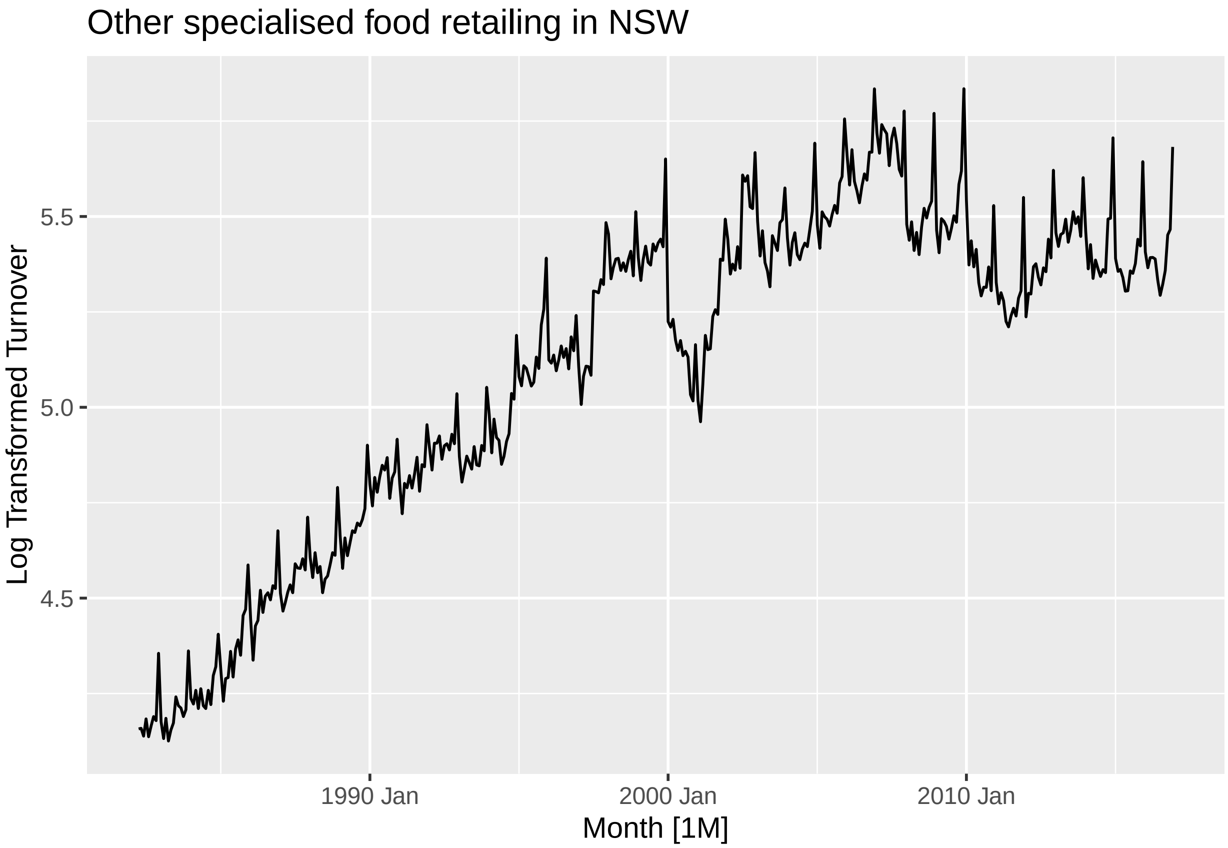
2. Transformation

It is known previously that the data is not stationary. Thus, to stabilize the variance, transformation needs to be done.

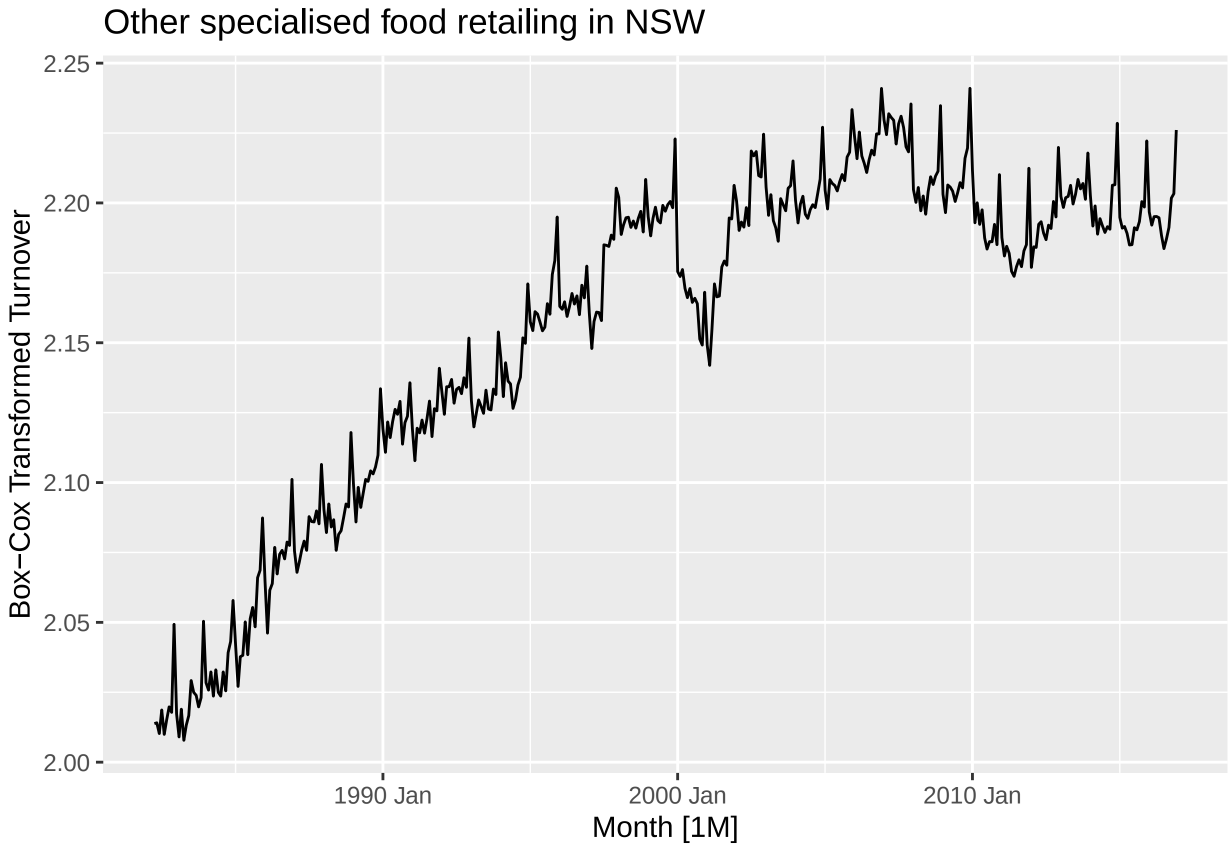
The appropriate λ to be chosen, is the λ in which the variation of the data seems constant over time. As the variation increases as Turnover increases, λ is supposed to be less than 1.

As chosen by the guerrero features, the λ to be chosen is -0.4039. This balance the seasonal fluctuation and random variation across the series. However, we still need to check again the resulting plot of the transformation, because sometimes lambda suggested by guerrero might not be very suitable for some cases. Compring the plot when data is transformed using box_cox and log, box_cox transformation performs better than a log transformation in making the variation of the data seems constant over time. Thus, box-cox transformation with $\lambda = -0.4039$ is chosen.

```
lambda <- myseries_tr %>% features(Turnover,features=guerrero)%>% pull(lambda_guerrero)

myseries_tr %>% autoplot(log(Turnover))+ylab("Log Transformed Turnover")+labs(title="Other specialised f
```



```
myseries_tr %>% autoplot(box_cox(Turnover,lambda))+ylab("Box-Cox Transformed Turnover")+labs(title="Othe
```

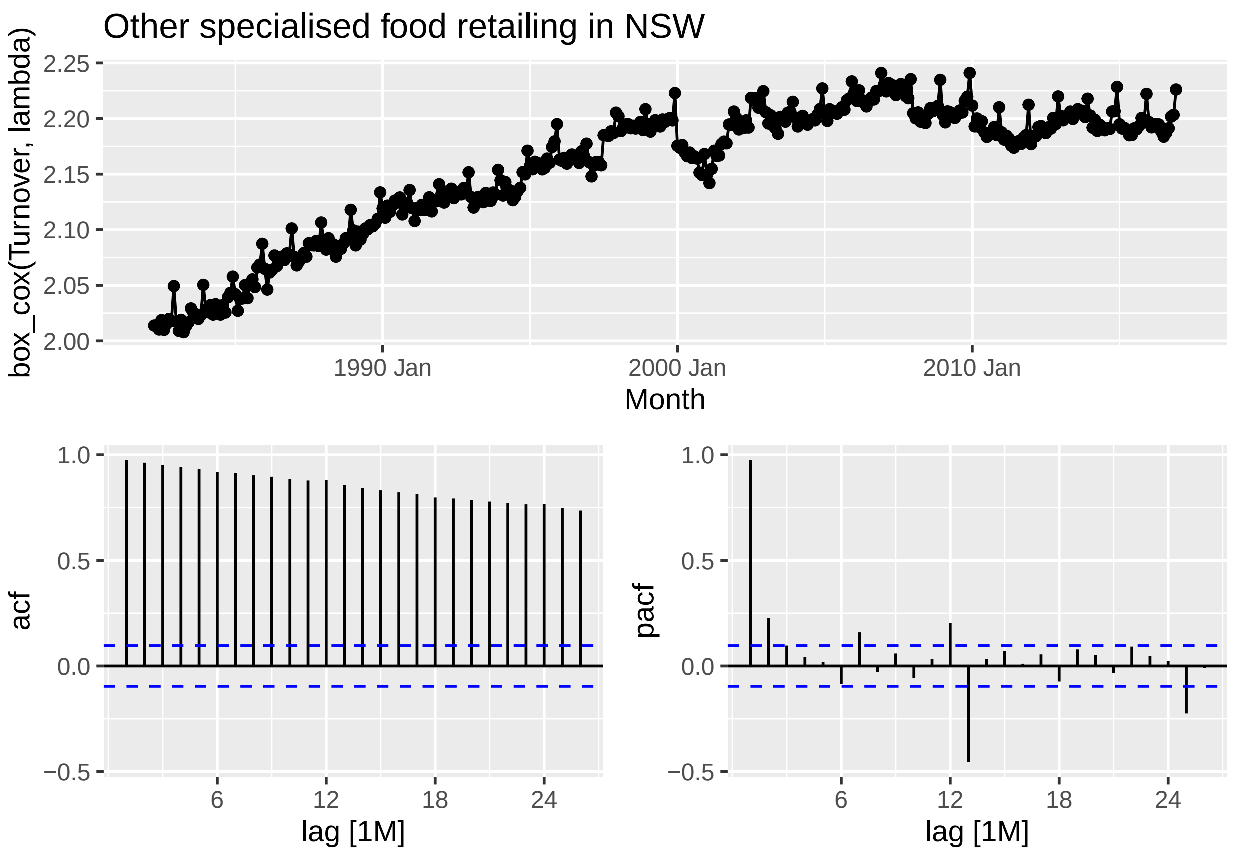


3. Checking for Differencing

After doing th transformation, the data is not yet stationary as the ACF is decreasing slowly. Thus, we need to the differencing, to stabilize the mean.

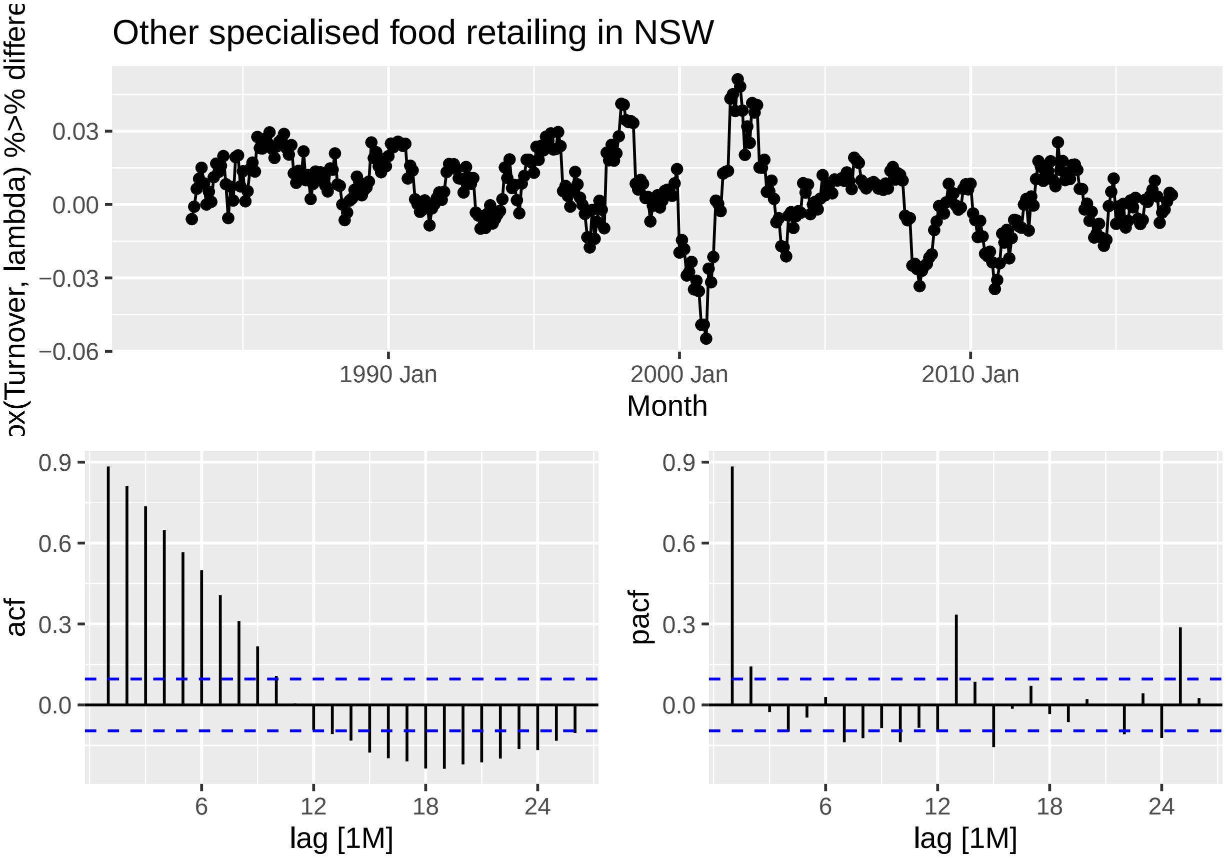
There are still some seasonality left in the transformed data, thus we might need to do seasoanal differencing. As this is monthly data, we will use lag=12. It could be argued that we might need another first difference to make the data stationary. This is beacuse, according to the ACF plot, the lag is still significant up to lag 9. Furthermore, we can reject the null that there is no unit root from the unit root test. After adding for the first differencing, the data looks stationary, as shown from the ACF, also we cannot reject the null that there is no unit root from the unit root test, at 5% level of significance.

```
myseries_tr %>% gg_tsdisplay(box_cox(Turnover,lambda), plot_type = "partial")+labs(title="Other speciali
```



```
#Seasonal differencing
myseries_tr %>% gg_tsdisplay((box_cox(Turnover,lambda) %>% difference(12)),plot_type="partial")+labs(ti
```

```
## Warning: Removed 12 row(s) containing missing values (geom_path).

## Warning: Removed 12 rows containing missing values (geom_point).
```



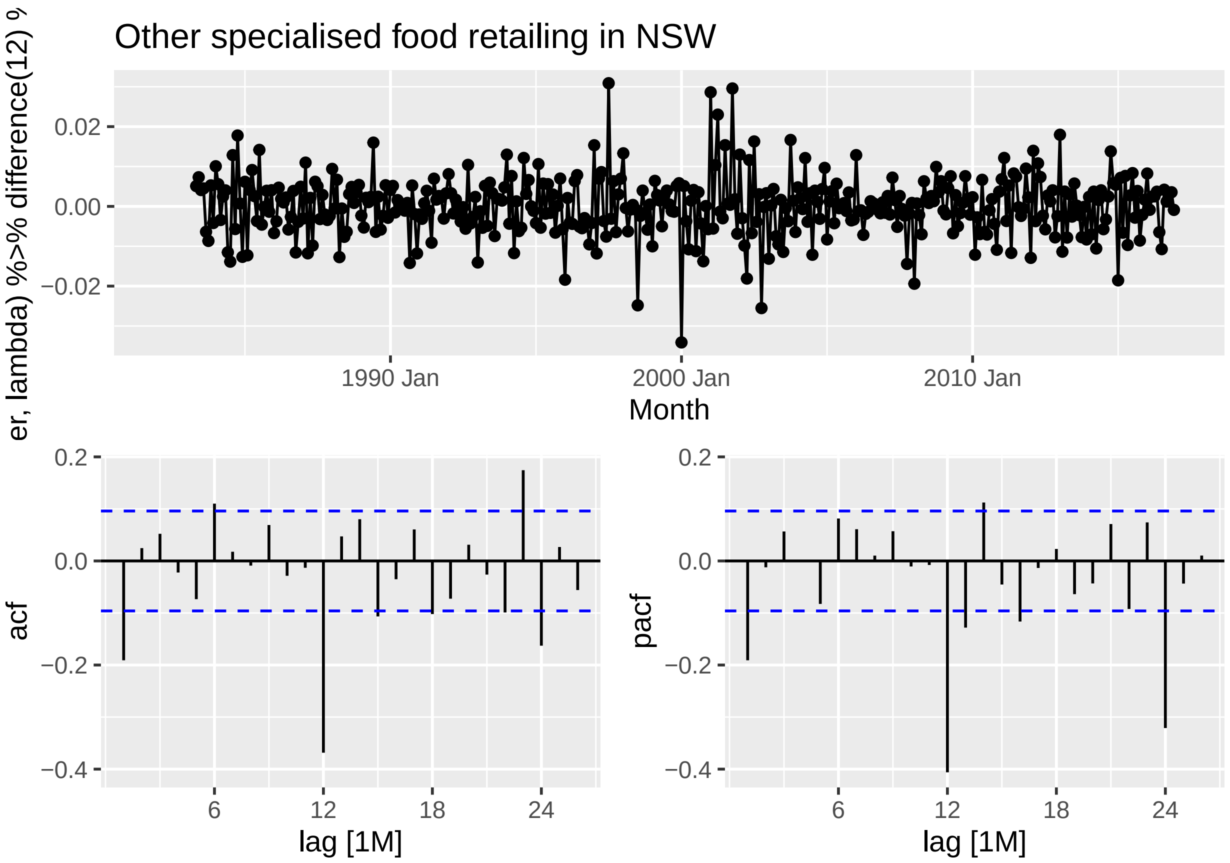
```
myseries_tr %>% mutate(diff=difference(box_cox(Turnover,lambda),12)) %>% features(diff,unitroot_kpss)
```

```
## # A tibble: 1 x 4
##   State           Industry                          kpss_stat kpss_pvalue
##   <chr>           <chr>                                 <dbl>       <dbl>
## 1 New South Wales Other specialised food retailing      0.962        0.01
```

```
#Adding first difference
myseries_tr %>% gg_tsdisplay((box_cox(Turnover,lambda) %>% difference(12)%>% difference(1)),plot_type="
```

```
## Warning: Removed 13 row(s) containing missing values (geom_path).
```

```
## Warning: Removed 13 rows containing missing values (geom_point).
```



```
myseries_tr %>% mutate(diff=difference(difference(box_cox(Turnover,lambda),12),1)) %>% features(diff,un
```

```
## # A tibble: 1 x 4
##   State           Industry                          kpss_stat kpss_pvalue
##   <chr>           <chr>                                 <dbl>       <dbl>
## 1 New South Wales Other specialised food retailing     0.0194         0.1
```

4. Fit the Model

It could be seen from the PACF that there is a spike up to lag 2, so p=2. Or, as showed by ACF plot there is a significant lag up to lag 2 so q=2 might be appropriate.We know previously that to make the data stationary, we take first differencing, so d=1 might be appropriate.

To account for the seasonal component, it could be seen from the PACF, that lag 12 and 24 are significant, so P=2 might be appropriate, or based on ACF plot, lag 12 and 24 are significant, so Q=2 might be appropriate. We know previously that we take seasonal differencing, so D=1 will be appropriate.

Thus, some ARIMA models that might be suitable are ARIMA(2,1,0)(2,1,0), ARIMA(2,1,0)(0,1,2), ARIMA(0,1,2)(2,1,0), and ARIMA(0,1,2)(0,1,2).

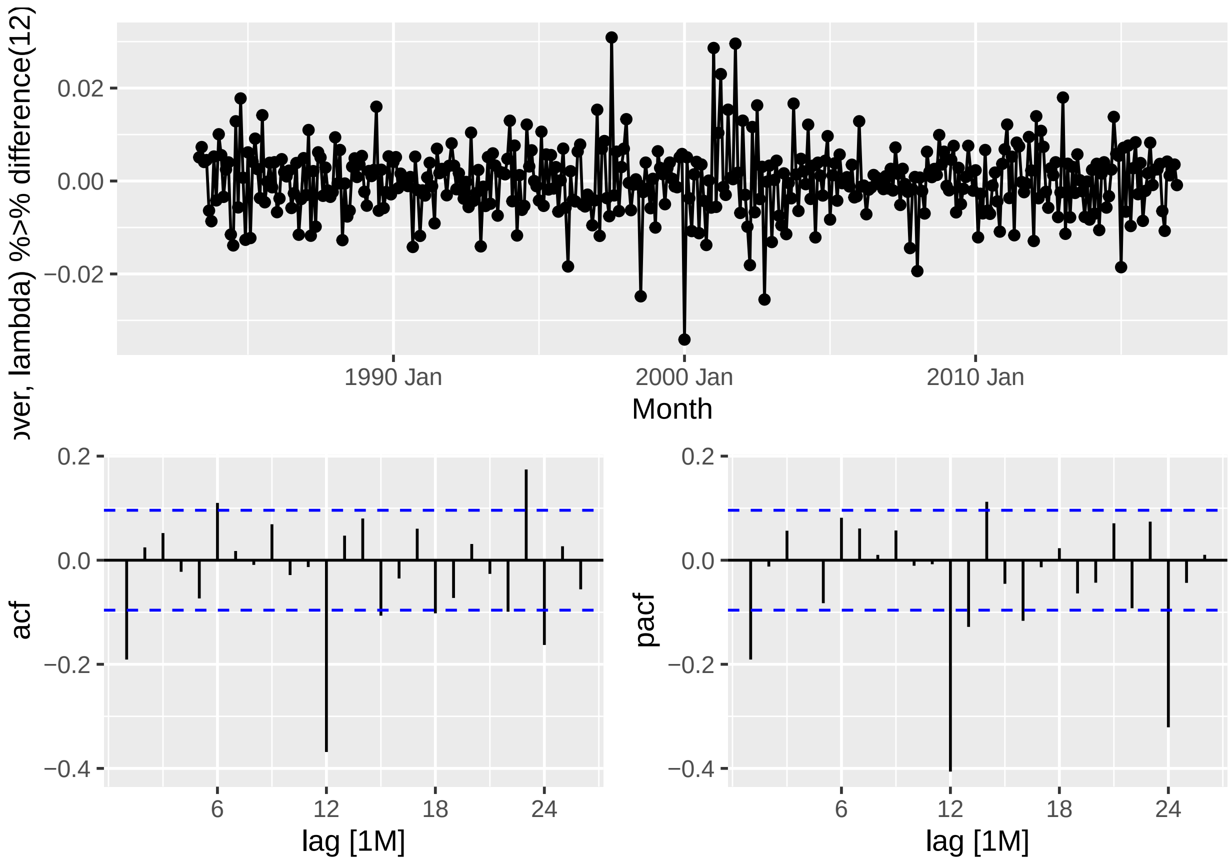
Just looking from the ACF and PACF, we can only determine p or q, P or Q, but not both.

It has been decided not to include a constant, as we have 2 differencing, so including a constant will make long-term forecast follow a quadratic trend. Having c=0 and having 2 differencing will make the long-term forecast follow a straight line.

As all of the models have the same number of differencing, AICc could be used to choose the more suitable model. It could be seen that the auto model has the lowest AICc, where auto model is ARIMA(0,1,2)(0,1,2).

```
myseries_tr%>% gg_tsdisplay((box_cox(Turnover,lambda) %>% difference(12)%>% difference(1)),plot_type="pa
```

```
## Warning: Removed 13 row(s) containing missing values (geom_path).

## Warning: Removed 13 rows containing missing values (geom_point).
```



```
arima <- myseries_tr%>% model(arima210210=ARIMA(box_cox(Turnover,lambda)~0+pdq(2,1,0)+PDQ(2,1,0)),
                              arima210012=ARIMA(box_cox(Turnover,lambda)~0+pdq(2,1,0)+PDQ(2,1,0)),
                              arima012210=ARIMA(box_cox(Turnover,lambda)~0+pdq(2,1,0)+PDQ(2,1,0)),
                              arima012012=ARIMA(box_cox(Turnover,lambda)~0+pdq(0,1,2)+PDQ(0,1,2)),
                              auto=ARIMA(box_cox(Turnover,lambda)))

arima %>% glance() %>% arrange(AICc) %>% select(State:AICc)
```

```
## # A tibble: 5 x 7
##   State           Industry                   .model  sigma2 log_lik    AIC   AICc
##   <chr>           <chr>                      <chr>    <dbl>   <dbl>  <dbl>  <dbl>
## 1 New South Wales Other specialised food r~ arima~ 3.12e-5   1513. -3017. -3017.
## 2 New South Wales Other specialised food r~ arima~ 4.00e-5   1472. -2935. -2935.
## 3 New South Wales Other specialised food r~ arima~ 4.00e-5   1472. -2935. -2935.
## 4 New South Wales Other specialised food r~ arima~ 4.00e-5   1472. -2935. -2935.
## 5 New South Wales Other specialised food r~ auto   4.46e-5   1452. -2892. -2892.
```

```r
best_arima <- myseries_tr %>%model(ARIMA(box_cox(Turnover,lambda)~0+pdq(0,1,2)+PDQ(0,1,2)))

best_arima %>% report()
```

```
## Series: Turnover
## Model: ARIMA(0,1,2)(0,1,2)[12]
## Transformation: box_cox(Turnover, lambda)
##
## Coefficients:
##           ma1     ma2     sma1     sma2
##       -0.2545  0.0882  -0.8427  -0.0917
## s.e.   0.0497  0.0545   0.0588   0.0611
##
## sigma^2 estimated as 3.121e-05:  log likelihood=1513.44
## AIC=-3016.88   AICc=-3016.73   BIC=-2996.88
```

```r
best_arima %>% tidy()
```

```
## # A tibble: 4 x 8
##   State           Industry    .model term  estimate std.error statistic  p.value
##   <chr>           <chr>       <chr>  <chr>    <dbl>     <dbl>     <dbl>    <dbl>
## 1 New South Wales Other spec~ "ARIM~ ma1    -0.254      0.0497     -5.12 4.74e- 7
## 2 New South Wales Other spec~ "ARIM~ ma2     0.0882     0.0545      1.62 1.06e- 1
## 3 New South Wales Other spec~ "ARIM~ sma1   -0.843      0.0588    -14.3  6.20e-38
## 4 New South Wales Other spec~ "ARIM~ sma2   -0.0917     0.0611     -1.50 1.35e- 1
```

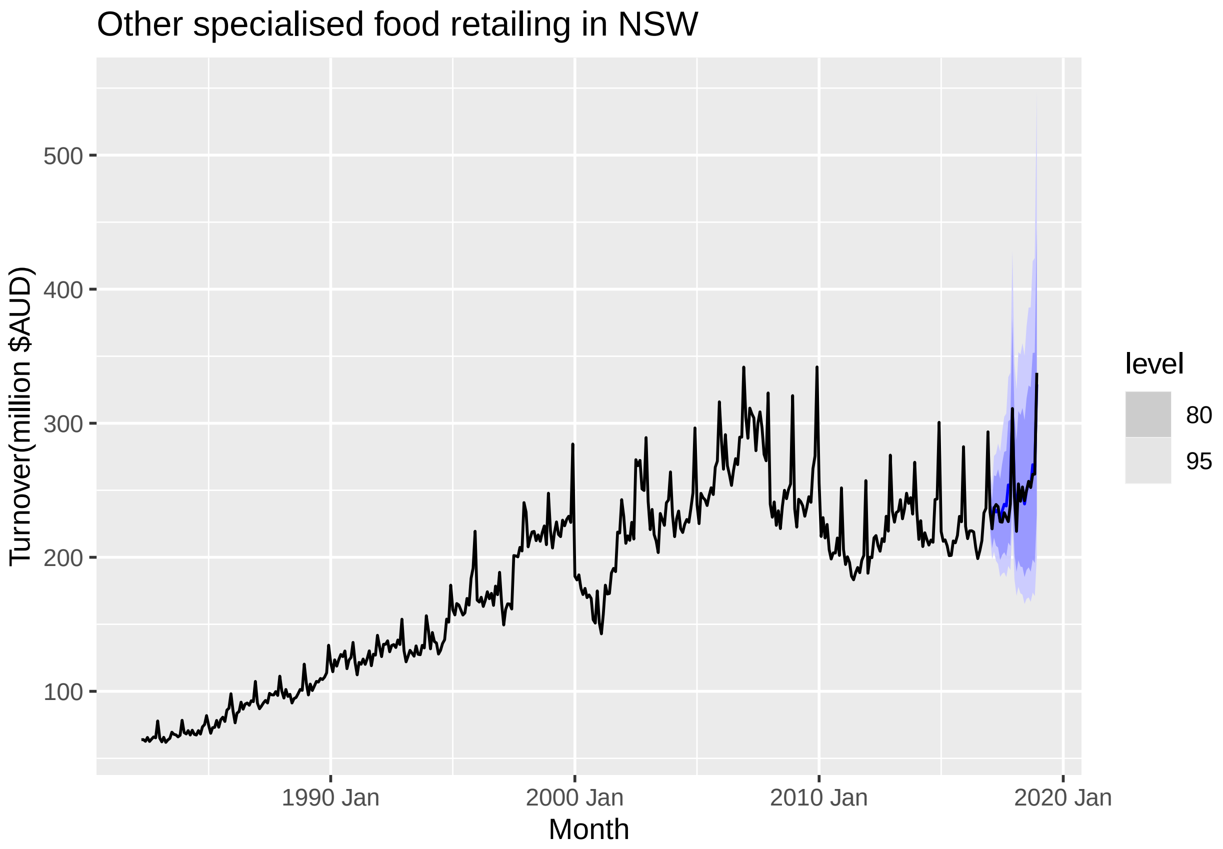
5. Forecast

Forecast for the last 24 months of the original data is produced, as well as its 80% prediction interval. The plot comparing th actual values and the forecasted values is shown below. It could be seen that the model has captured the the information well, as the forecasted value is quite similar to the actual values.

```r
test_fc_arima <- best_arima%>% forecast(h="24 months")

interval_arima <- test_fc_arima %>% mutate(interval=hilo(Turnover,0.80)) %>% pull(interval)
test_fc_arima <- test_fc_arima %>% mutate(Interval=interval_arima)
test_fc_arima
```

```
## # A fable: 24 x 7 [1M]
## # Key:     State, Industry, .model [1]
##    State           Industry             .model    Month            Turnover .mean
##    <chr>           <chr>                <chr>     <mth>              <dist> <dbl>
##  1 New South Wales Other specialised f~ "ARIM~ 2017 Jan t(N(2.2, 3.1e-05))  238.
##  2 New South Wales Other specialised f~ "ARIM~ 2017 Feb t(N(2.2, 4.9e-05))  224.
##  3 New South Wales Other specialised f~ "ARIM~ 2017 Mar   t(N(2.2, 7e-05))  237.
##  4 New South Wales Other specialised f~ "ARIM~ 2017 Apr t(N(2.2, 9.2e-05))  234.
##  5 New South Wales Other specialised f~ "ARIM~ 2017 May t(N(2.2, 0.00011))  235.
##  6 New South Wales Other specialised f~ "ARIM~ 2017 Jun t(N(2.2, 0.00014))  227.
##  7 New South Wales Other specialised f~ "ARIM~ 2017 Jul t(N(2.2, 0.00016))  234.
##  8 New South Wales Other specialised f~ "ARIM~ 2017 Aug t(N(2.2, 0.00018))  240.
##  9 New South Wales Other specialised f~ "ARIM~ 2017 Sep   t(N(2.2, 2e-04))  238.
## 10 New South Wales Other specialised f~ "ARIM~ 2017 Oct t(N(2.2, 0.00022))  254.
## # ... with 14 more rows, and 1 more variable: Interval <hilo>
```

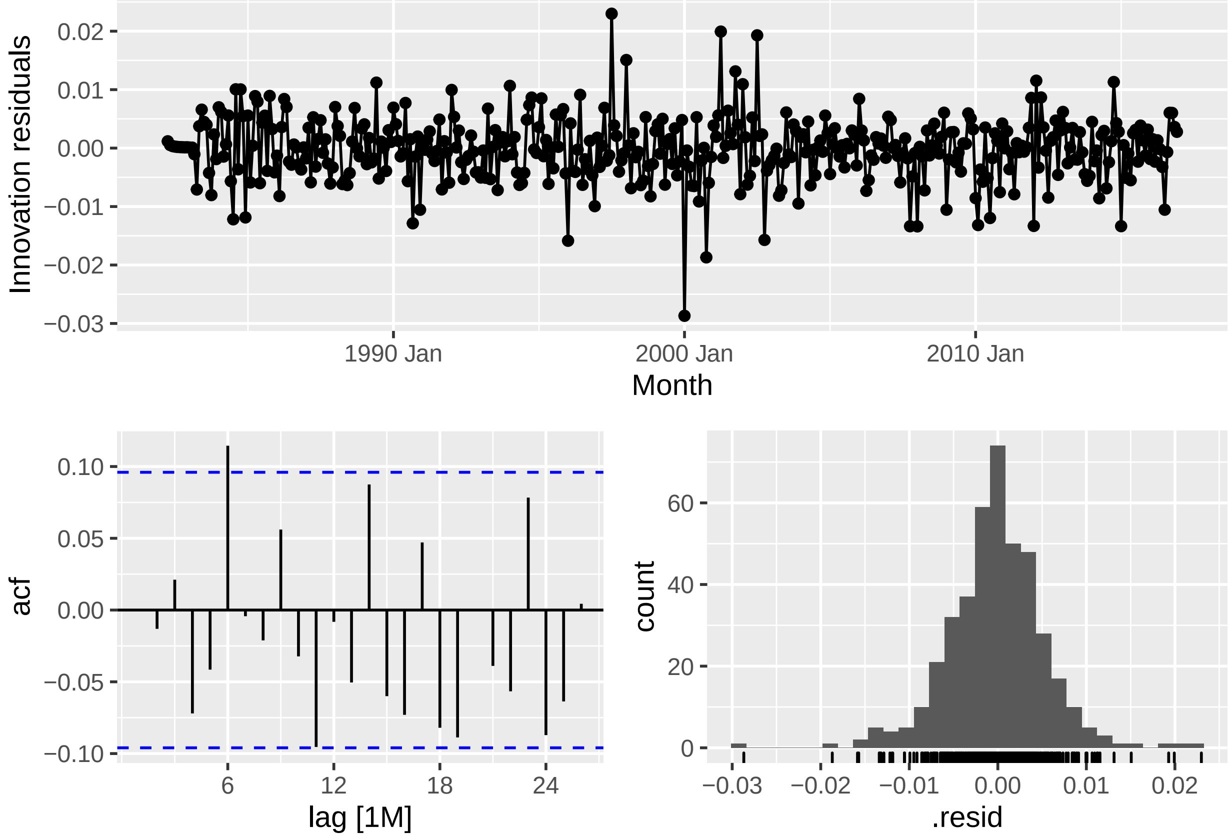
```
test_fc_arima %>% autoplot(myseries)+labs(title="Other specialised food retailing in NSW",y="Turnover(mi
```



6. Residual Diagnostics

There are still a little bit of autocorrelation in the residual as shown by the ACF, as it is significant at lag 6. From the ljung-box test(pvalue=0.007), it could be seen that we can reject the null that there is no autocorrelation in residual at 5% level of significance.

```
best_arima %>% gg_tsresiduals()
```



```
best_arima%>%
  augment() %>%
  features(.innov, ljung_box, dof = 4, lag = 24)
```

```
## # A tibble: 1 x 5
##   State           Industry                          .model      lb_stat lb_pvalue
##   <chr>           <chr>                             <chr>         <dbl>     <dbl>
## 1 New South Wales Other specialised food retailing "ARIMA(box~     38.6   0.00756
```

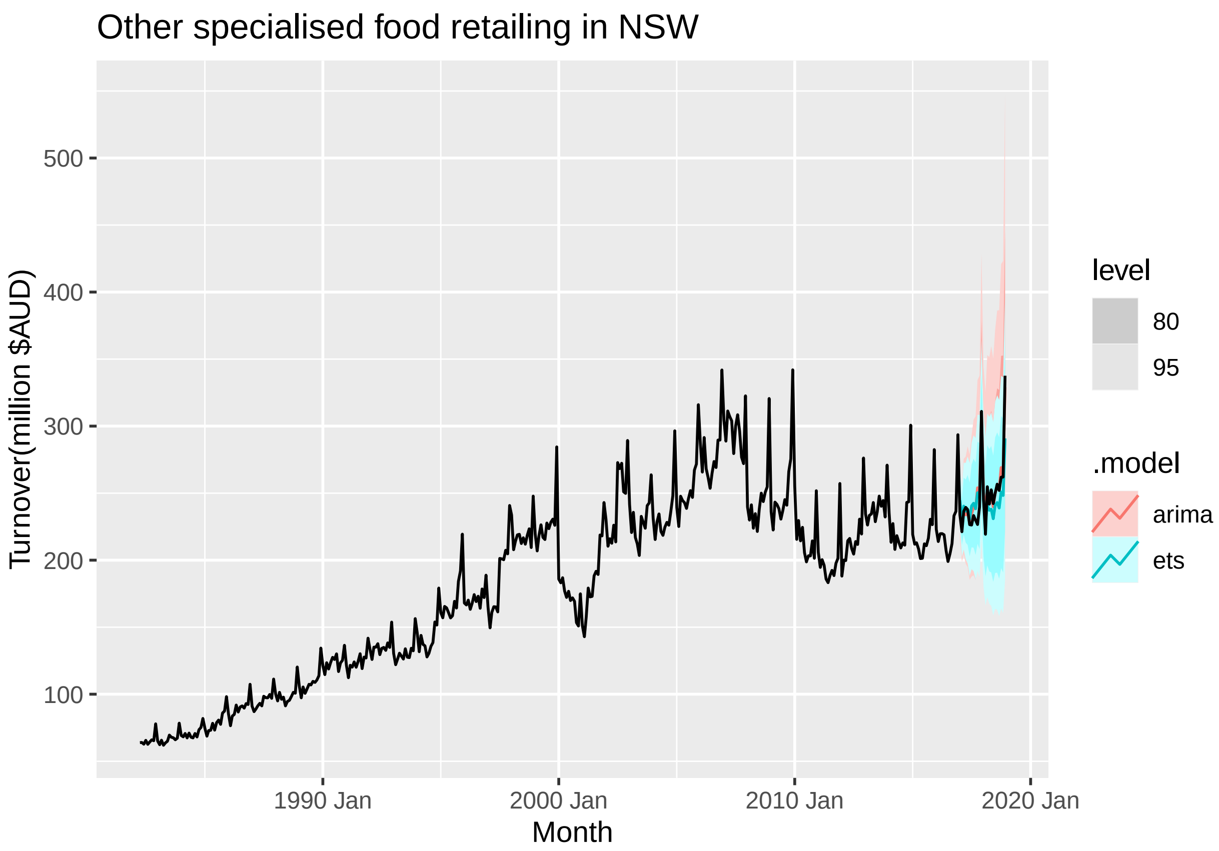
C. Comparing ETS and ARIMA

As could be seen, ARIMA(0,1,2)(0,1,2) model has better accuracy in forecasting the 24 months, as it has lower RMSE and MASE compared to ETS(M,Ad,M) model. Furthermore, it could be seen that the forecast produced by ARIMA(0,1,2)(0,1,2) has wider prediction interval than the ETS model, which means it captures the uncertainty about the future Therefore, ARIMA(0,1,2)(0,1,2) model is preferred.

```
model_combined <- myseries_tr %>% model(ets=ETS(Turnover),arima=ARIMA(box_cox(Turnover,lambda)~0+pdq(0,

fc_combined <- model_combined %>% forecast(h="24 months")
fc_combined %>% accuracy(myseries) %>% arrange(RMSE)
```

```
## # A tibble: 2 x 12
##   .model State   Industry .type    ME  RMSE   MAE    MPE  MAPE  MASE RMSSE  ACF1
##   <chr>  <chr>   <chr>    <chr> <dbl> <dbl> <dbl>  <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 arima  New So~ Other s~ Test  -3.40  8.42  5.99 -1.53   2.50 0.318 0.317 0.278
## 2 ets    New So~ Other s~ Test   3.15 14.7  11.5   0.813  4.45 0.608 0.555 0.392
```

```
model_combined %>% forecast(h="24 months") %>% autoplot(myseries) +labs(title="Other specialised food r
```



Forecasting Next 2 Years Data

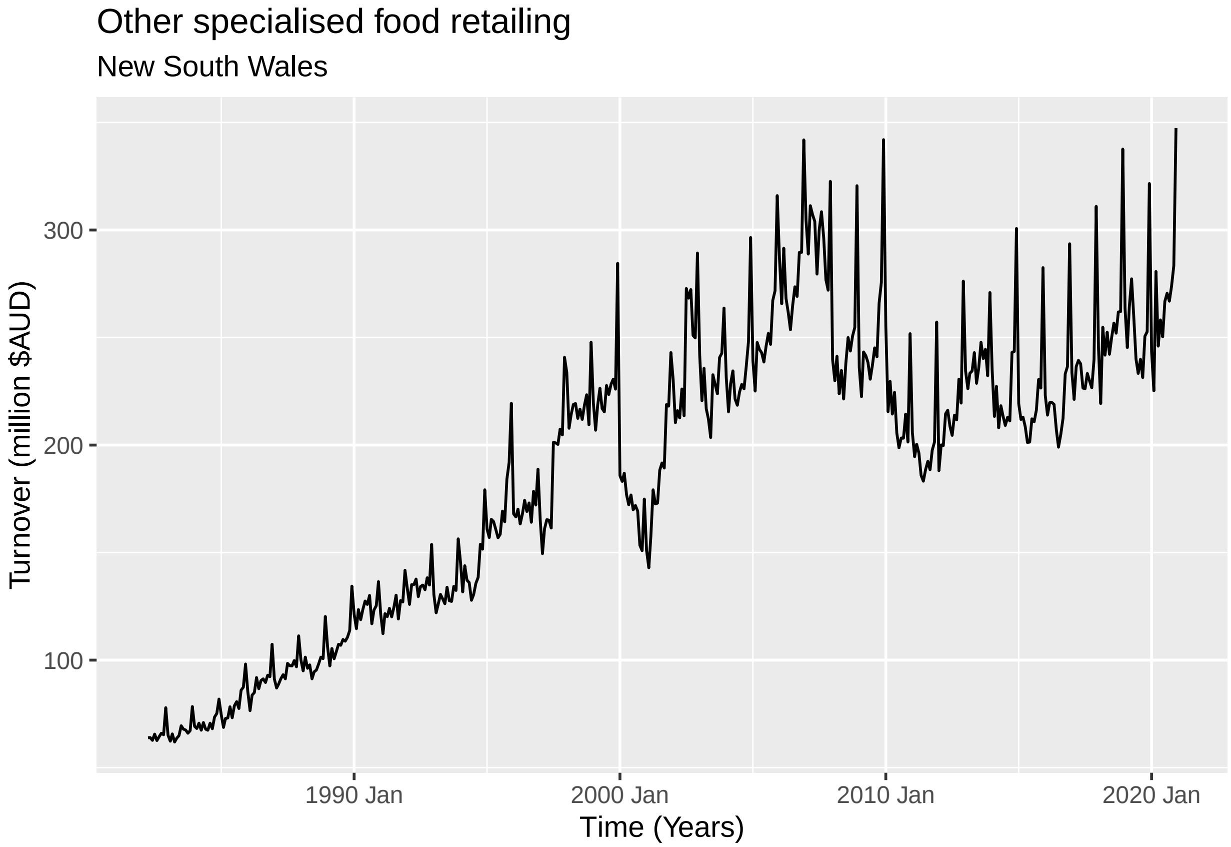
```
new_data <- read.csv("abs_new.csv") %>% mutate(Month=yearmonth(Month))
colnames(new_data)[1] <- "State"
new_data <- new_data %>% mutate(Month=yearmonth(Month))%>% as_tsibble(key=c(State,Industry))
```

```
## Using ‘Month‘ as index variable.
```

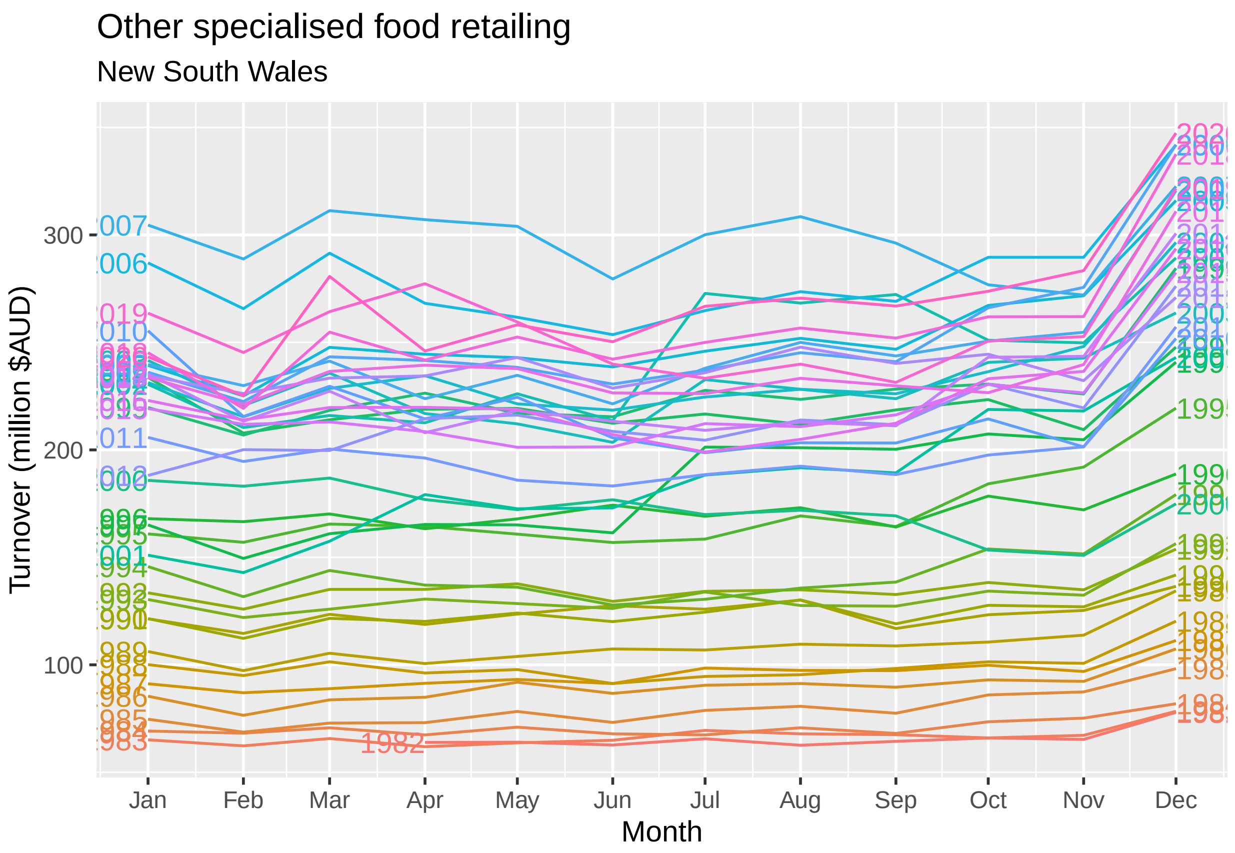
Overtime, we could see there is an upward trend, although there has been a lot of volatility in the data. Turnover tend to fall in February and increase in December which is its highest turnover for the year. The magnitude of the changes in turnover seems to be more volatile for the more recent years.

This seasonality pattern observed previously from seasonality graph also supported by graph of the subseries, where it could be seen that average of the turnover for December across the years is the highest compared to other months, as well as February having the lowest average turnover across the years. Turnover across years have been increasing for each month, on average.

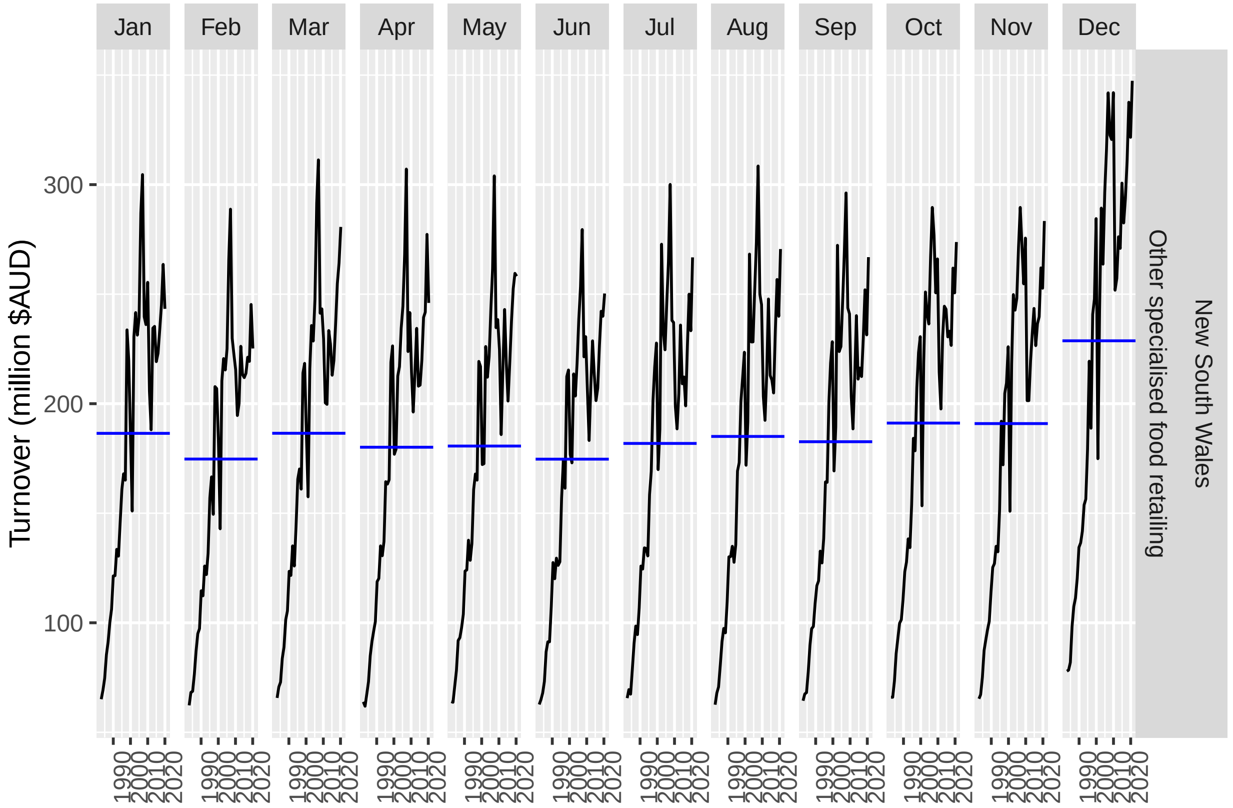
```
new_data %>% autoplot(Turnover) +
  labs(y = "Turnover (million $AUD)", x = "Time (Years)",
       title = myseries$Industry[1],
       subtitle = myseries$State[1])
```



```
new_data %>%
  gg_season(Turnover, labels = "both") +
  labs(y = "Turnover (million $AUD)",
       title = myseries$Industry[1],
       subtitle = myseries$State[1])
```



```
new_data %>%
  gg_subseries(Turnover) +
  labs(y = "Turnover (million $AUD)", x="")
```



A. ETS model

1. Fit the model

```
myseries_n <- myseries %>% select(-`Series ID`)
best_ets_n <- myseries_n%>% model(ETS(Turnover~error("M")+trend("Ad")+season("M")))

report(best_ets_n)
```

```
## Series: Turnover
## Model: ETS(M,Ad,M)
##   Smoothing parameters:
##     alpha = 0.7691719
##     beta  = 0.001403467
##     gamma = 0.05620414
##     phi   = 0.9799998
##
##   Initial states:
##       l[0]      b[0]      s[0]      s[-1]     s[-2]     s[-3]     s[-4]    s[-5]
##   64.21472 0.3647063 0.990008 0.9377055 1.003333 1.168349 1.020121 1.02919
##       s[-6]     s[-7]      s[-8]      s[-9]     s[-10]     s[-11]
##   0.9839443 0.998651 0.9915236 0.9430394 0.9755594 0.9585759
```

```
##
##   sigma^2:  0.0021
##
##      AIC     AICc      BIC
## 4507.694 4509.315 4581.296
```

```
tidy(best_ets_n)
```

```
## # A tibble: 18 x 5
##    State           Industry                         .model        term  estimate
##    <chr>           <chr>                            <chr>         <chr>    <dbl>
##  1 New South Wales Other specialised food retailing "ETS(Turnove~ alpha  0.769
##  2 New South Wales Other specialised food retailing "ETS(Turnove~ beta   0.00140
##  3 New South Wales Other specialised food retailing "ETS(Turnove~ gamma  0.0562
##  4 New South Wales Other specialised food retailing "ETS(Turnove~ phi    0.980
##  5 New South Wales Other specialised food retailing "ETS(Turnove~ l[0]  64.2
##  6 New South Wales Other specialised food retailing "ETS(Turnove~ b[0]   0.365
##  7 New South Wales Other specialised food retailing "ETS(Turnove~ s[0]   0.990
##  8 New South Wales Other specialised food retailing "ETS(Turnove~ s[-1]  0.938
##  9 New South Wales Other specialised food retailing "ETS(Turnove~ s[-2]  1.00
## 10 New South Wales Other specialised food retailing "ETS(Turnove~ s[-3]  1.17
## 11 New South Wales Other specialised food retailing "ETS(Turnove~ s[-4]  1.02
## 12 New South Wales Other specialised food retailing "ETS(Turnove~ s[-5]  1.03
## 13 New South Wales Other specialised food retailing "ETS(Turnove~ s[-6]  0.984
## 14 New South Wales Other specialised food retailing "ETS(Turnove~ s[-7]  0.999
## 15 New South Wales Other specialised food retailing "ETS(Turnove~ s[-8]  0.992
## 16 New South Wales Other specialised food retailing "ETS(Turnove~ s[-9]  0.943
## 17 New South Wales Other specialised food retailing "ETS(Turnove~ s[-1~  0.976
## 18 New South Wales Other specialised food retailing "ETS(Turnove~ s[-1~  0.959
```

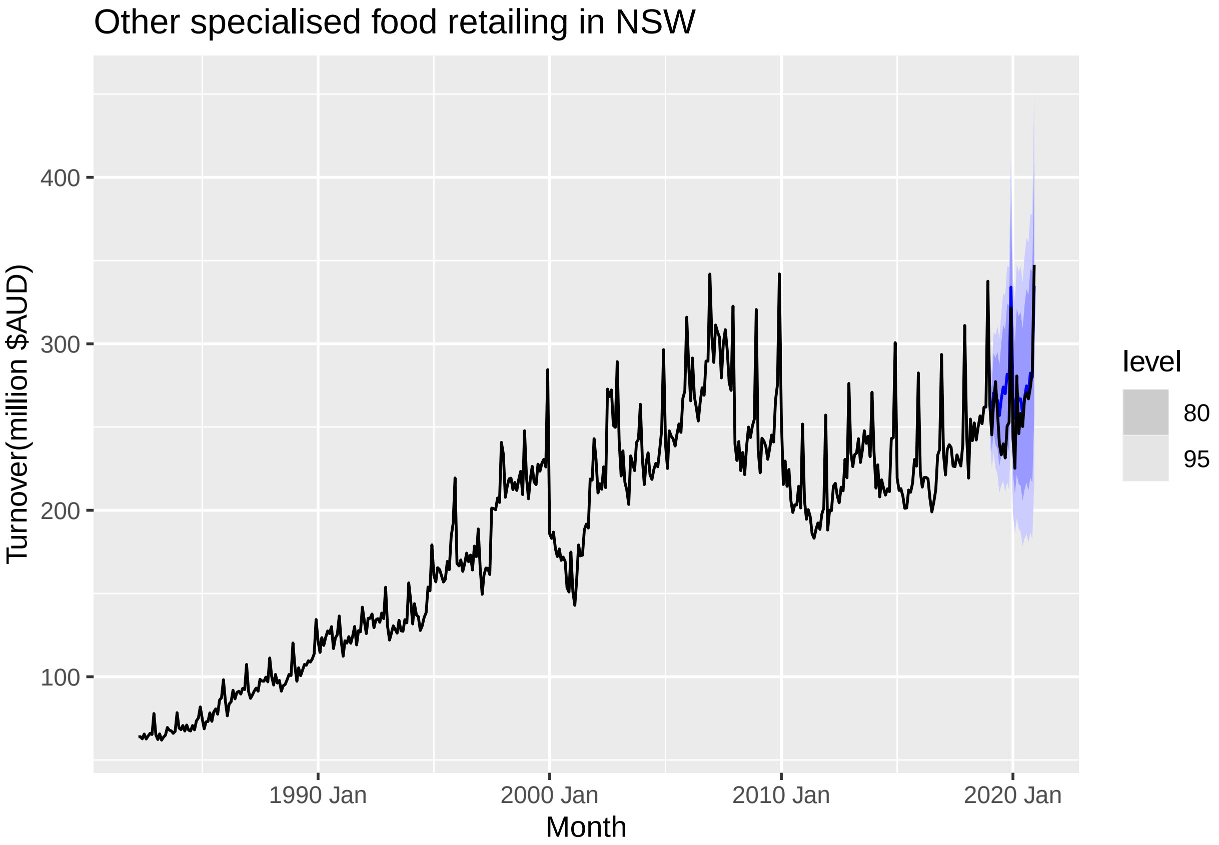
2. Produce Forecast

Forecast for the next 2 years is produced. The plot comparing th actual values and the forecasted values is shown below, as well as its 80% prediction interval. It could be seen that the model has captured the the information well, as the forecasted value is quite similar to the actual values.

```
test_fc_ets_n <- best_ets_n %>% forecast(h="2 years")
interval_n <- test_fc_ets_n %>% mutate(interval=hilo(Turnover,0.80)) %>% pull(interval)
test_fc_ets_n <- test_fc_ets_n %>% mutate(Interval=interval)
test_fc_ets_n
```

```
## # A fable: 24 x 7 [1M]
## # Key:     State, Industry, .model [1]
##    State     Industry .model    Month     Turnover .mean                 Interval
##    <chr>     <chr>    <chr>     <mth>       <dist> <dbl>                   <hilo>
##  1 New Sout~ Other s~ "ETS(~ 2019 Jan  N(268, 152)  268. [244.5254, 244.7508]0.8
##  2 New Sout~ Other s~ "ETS(~ 2019 Feb  N(255, 220)  255. [228.3803, 228.6478]0.8
##  3 New Sout~ Other s~ "ETS(~ 2019 Mar  N(271, 341)  271. [240.1632, 240.4941]0.8
##  4 New Sout~ Other s~ "ETS(~ 2019 Apr  N(265, 418)  265. [236.4342, 236.8025]0.8
##  5 New Sout~ Other s~ "ETS(~ 2019 May  N(266, 512)  266. [237.2509, 237.6592]0.8
##  6 New Sout~ Other s~ "ETS(~ 2019 Jun  N(257, 561)  257. [230.1834, 230.6139]0.8
##  7 New Sout~ Other s~ "ETS(~ 2019 Jul  N(267, 698)  267. [240.2435, 240.7264]0.8
##  8 New Sout~ Other s~ "ETS(~ 2019 Aug  N(274, 833)  274. [242.1898, 242.7084]0.8
##  9 New Sout~ Other s~ "ETS(~ 2019 Sep  N(270, 904)  270. [238.1250, 238.6645]0.8
## 10 New Sout~ Other s~ "ETS(~ 2019 Oct N(282, 1088)  282. [250.0078, 250.6039]0.8
## # ... with 14 more rows
```

```
best_ets_n %>% forecast(h="2 years") %>% autoplot(new_data)+labs(title="Other specialised food retailin
```



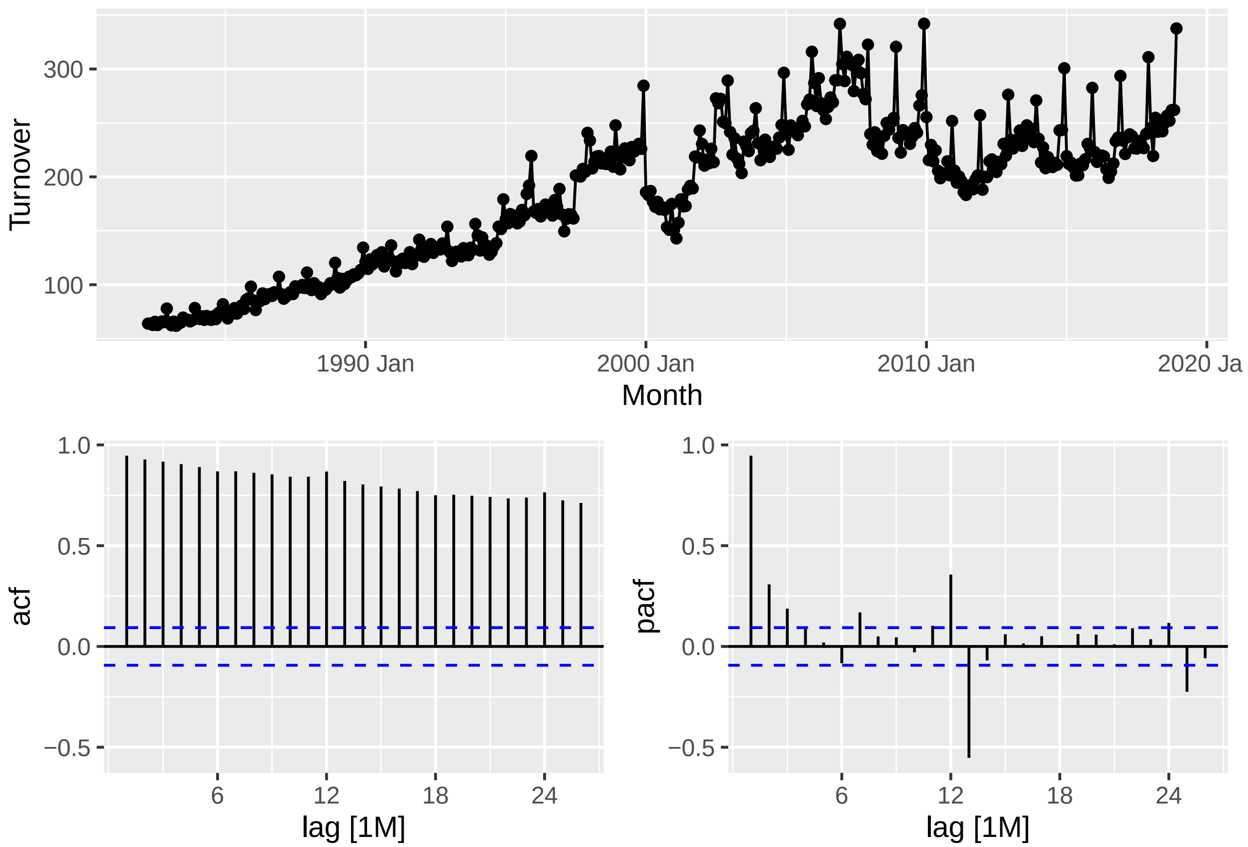
B. ARIMA Model

1. Checking for Stationarity

For ARIMA model, it is important that the data to be fitted is stationary. It could be seen from the graph, the existence of the upward trend and seasonality in the data. Thus, we can conclude that the data is not stationary. Furthermore, unit root test is also performed. As pvalue(0.01)<0.05, we can reject the null that there is no unit root, at 5% level of significance. Thus, we also conclude the data is not stationary.

```
myseries_n %>% gg_tsdisplay(plot_type="partial")
```

```
## Plot variable not specified, automatically selected ‘y = Turnover‘
```



```
myseries_n %>% features(Turnover,unitroot_kpss)
```

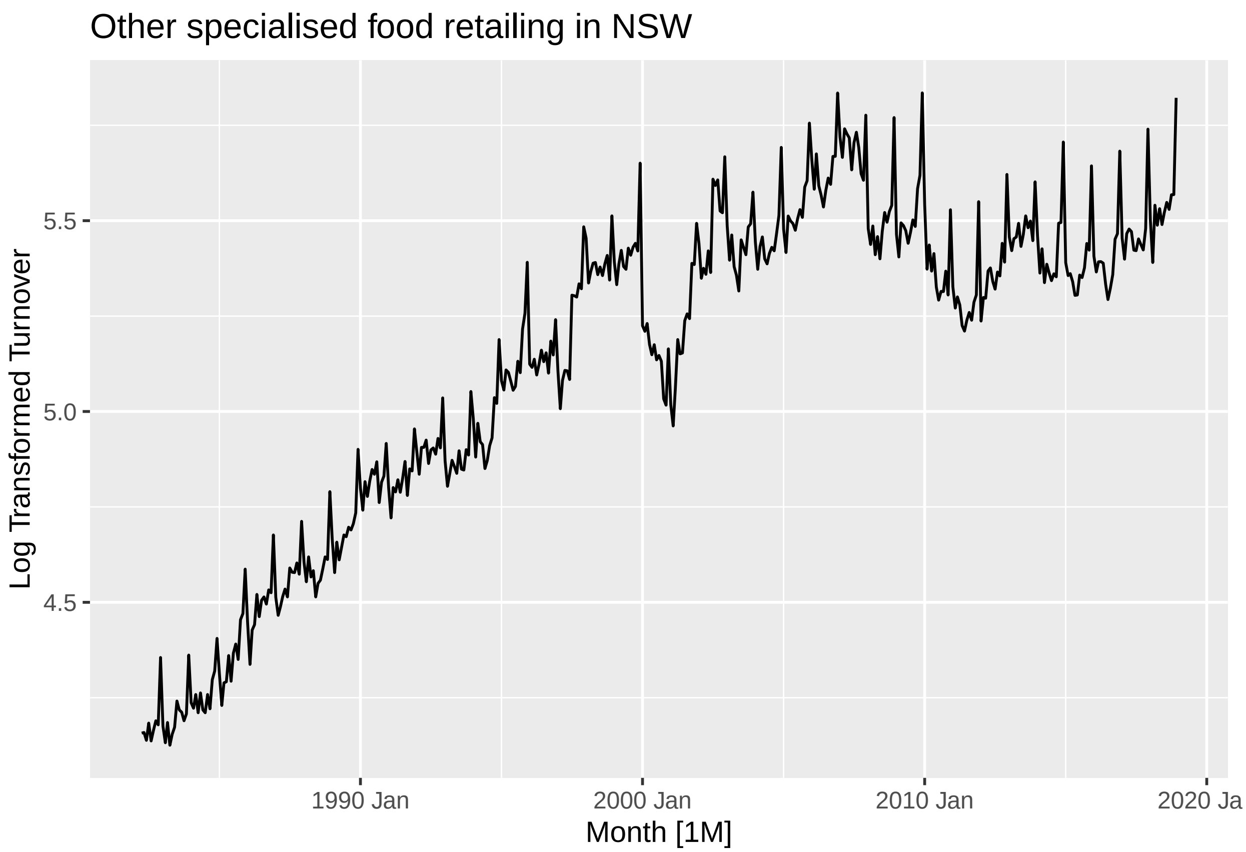
```
## # A tibble: 1 x 4
##   State           Industry                         kpss_stat kpss_pvalue
##   <chr>           <chr>                                <dbl>       <dbl>
## 1 New South Wales Other specialised food retailing      6.07        0.01
```

2. Transformation

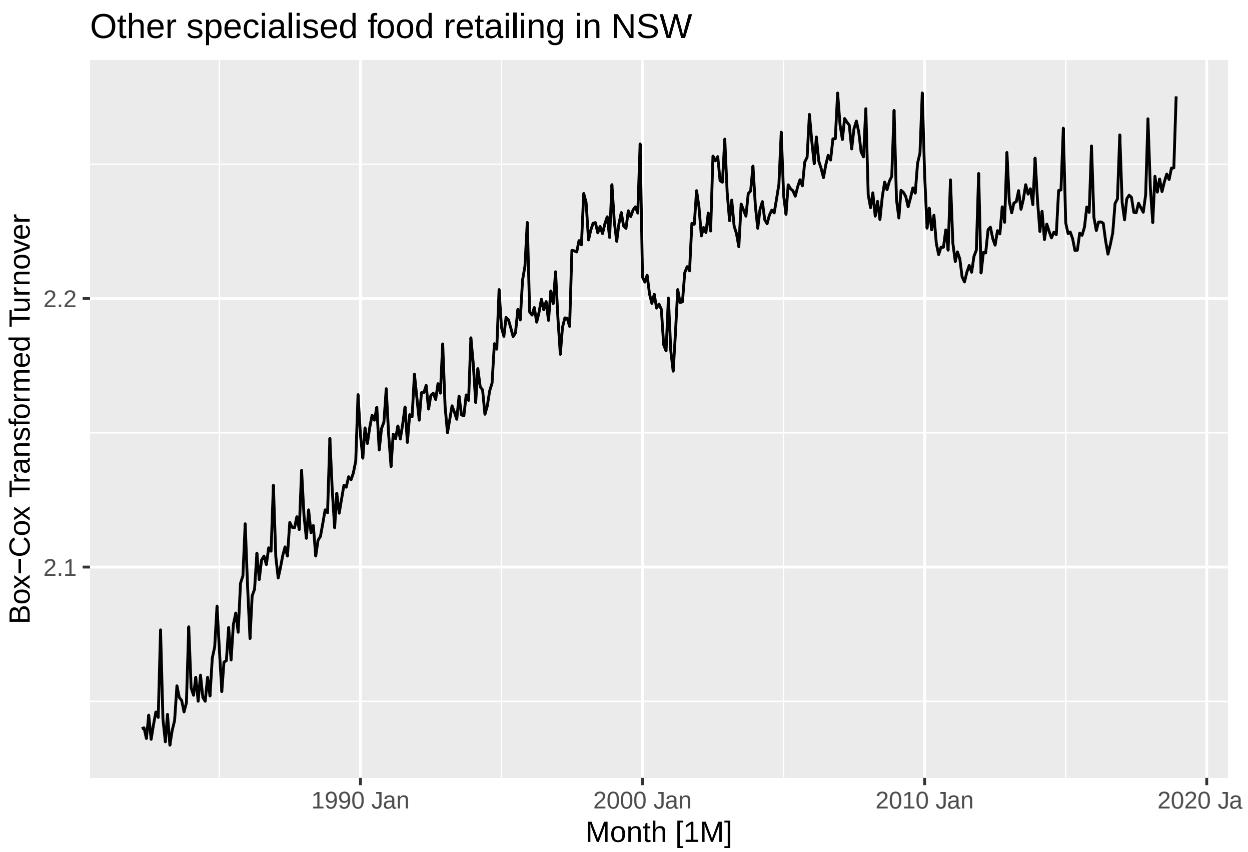
The appropriate λ to be chosen, is the λ in which the variation of the data seems constant over time. As the variation increases as Turnover increases, λ is supposed to be less than 1.

As chosen by the guerrero features, the λ to be chosen is -0.3955691. This balance the seasonal fluctuation and random variation across the series. However, we still need to check again the resulting plot of the transformation, because sometimes lambda suggested by guerrero might not be very suitable for some cases. However, in this case, this Box-Cox transformation performs better than a log transformation in making the variation of the data seems constant over time. Thus, box-cox transformation with $\lambda = -0.3955691$ is chosen.

```
lambda_n <- myseries %>% features(Turnover,features=guerrero)%>% pull(lambda_guerrero)

myseries_n %>% autoplot(log(Turnover))+ylab("Log Transformed Turnover")+labs(title="Other specialised fo
```



```
myseries_n %>% autoplot(box_cox(Turnover,lambda_n))+ylab("Box-Cox Transformed Turnover")+labs(title="Ot
```



3. Fit the Model

```
best_arima_n <- myseries_n %>%model(ARIMA(box_cox(Turnover,lambda_n)~0+pdq(0,1,2)+PDQ(0,1,2)))

best_arima_n %>% report()
```

```
## Series: Turnover
## Model: ARIMA(0,1,2)(0,1,2)[12]
## Transformation: box_cox(Turnover, lambda_n)
##
## Coefficients:
##           ma1     ma2     sma1     sma2
##       -0.2586  0.0792  -0.8391  -0.1035
## s.e.   0.0485  0.0525   0.0563   0.0591
##
## sigma^2 estimated as 3.284e-05:  log likelihood=1592.26
## AIC=-3174.53   AICc=-3174.38   BIC=-3154.23
```

```
best_arima_n %>% tidy()
```

```
## # A tibble: 4 x 8
##   State           Industry    .model term  estimate std.error statistic  p.value
##   <chr>           <chr>       <chr>  <chr>    <dbl>     <dbl>     <dbl>    <dbl>
## 1 New South Wales Other spec~ "ARIM~ ma1    -0.259     0.0485     -5.33 1.56e- 7
## 2 New South Wales Other spec~ "ARIM~ ma2     0.0792    0.0525      1.51 1.32e- 1
## 3 New South Wales Other spec~ "ARIM~ sma1   -0.839     0.0563    -14.9  9.84e-41
## 4 New South Wales Other spec~ "ARIM~ sma2   -0.104     0.0591     -1.75 8.06e- 2
```

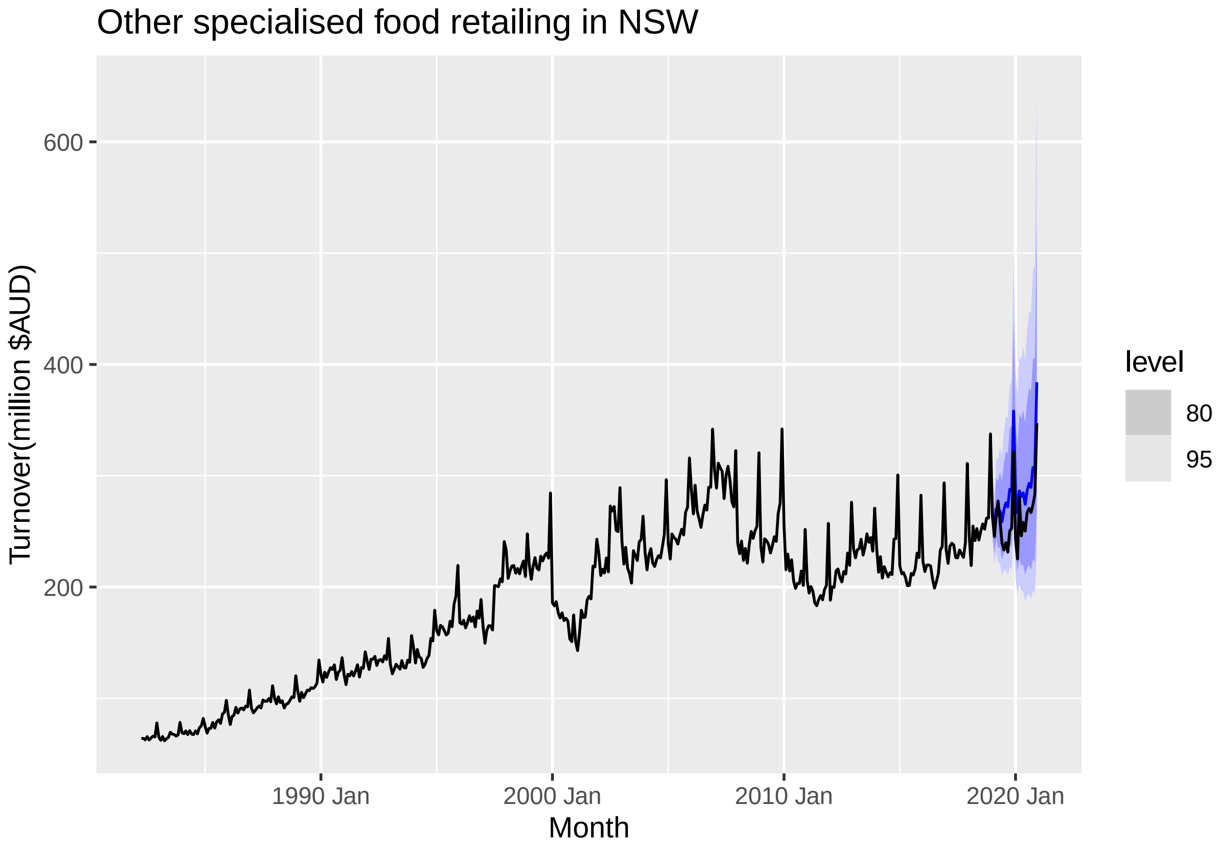
4. Forecast

Forecast for the next 2 years is produced. The plot comparing th actual values and the forecasted values is shown below, as well as its 80% prediction interval. It could be seen that the model has captured the the information well, as the forecasted value is quite similar to the actual values.

```
test_fc_arima_n <- best_arima_n%>% forecast(h="2 years")

interval_arima_n <- test_fc_arima_n %>% mutate(interval=hilo(Turnover,0.8)) %>% pull(interval)
test_fc_arima_n <- test_fc_arima_n %>% mutate(Interval=interval_arima_n)
test_fc_arima_n
```

```
## # A fable: 24 x 7 [1M]
## # Key:     State, Industry, .model [1]
##    State           Industry               .model    Month           Turnover .mean
##    <chr>           <chr>                  <chr>     <mth>             <dist> <dbl>
##  1 New South Wales Other specialised f~ "ARIM~ 2019 Jan t(N(2.3, 3.3e-05))  269.
##  2 New South Wales Other specialised f~ "ARIM~ 2019 Feb t(N(2.2, 5.1e-05))  250.
##  3 New South Wales Other specialised f~ "ARIM~ 2019 Mar t(N(2.3, 7.3e-05))  270.
##  4 New South Wales Other specialised f~ "ARIM~ 2019 Apr t(N(2.2, 9.5e-05))  264.
##  5 New South Wales Other specialised f~ "ARIM~ 2019 May t(N(2.3, 0.00012))  268.
##  6 New South Wales Other specialised f~ "ARIM~ 2019 Jun t(N(2.2, 0.00014))  259.
##  7 New South Wales Other specialised f~ "ARIM~ 2019 Jul t(N(2.3, 0.00016))  269.
##  8 New South Wales Other specialised f~ "ARIM~ 2019 Aug t(N(2.3, 0.00018))  276.
##  9 New South Wales Other specialised f~ "ARIM~ 2019 Sep t(N(2.3, 0.00021))  272.
## 10 New South Wales Other specialised f~ "ARIM~ 2019 Oct t(N(2.3, 0.00023))  288.
## # ... with 14 more rows, and 1 more variable: Interval <hilo>
```

```
test_fc_arima_n %>% autoplot(new_data)+labs(title="Other specialised food retailing in NSW",y="Turnover
```



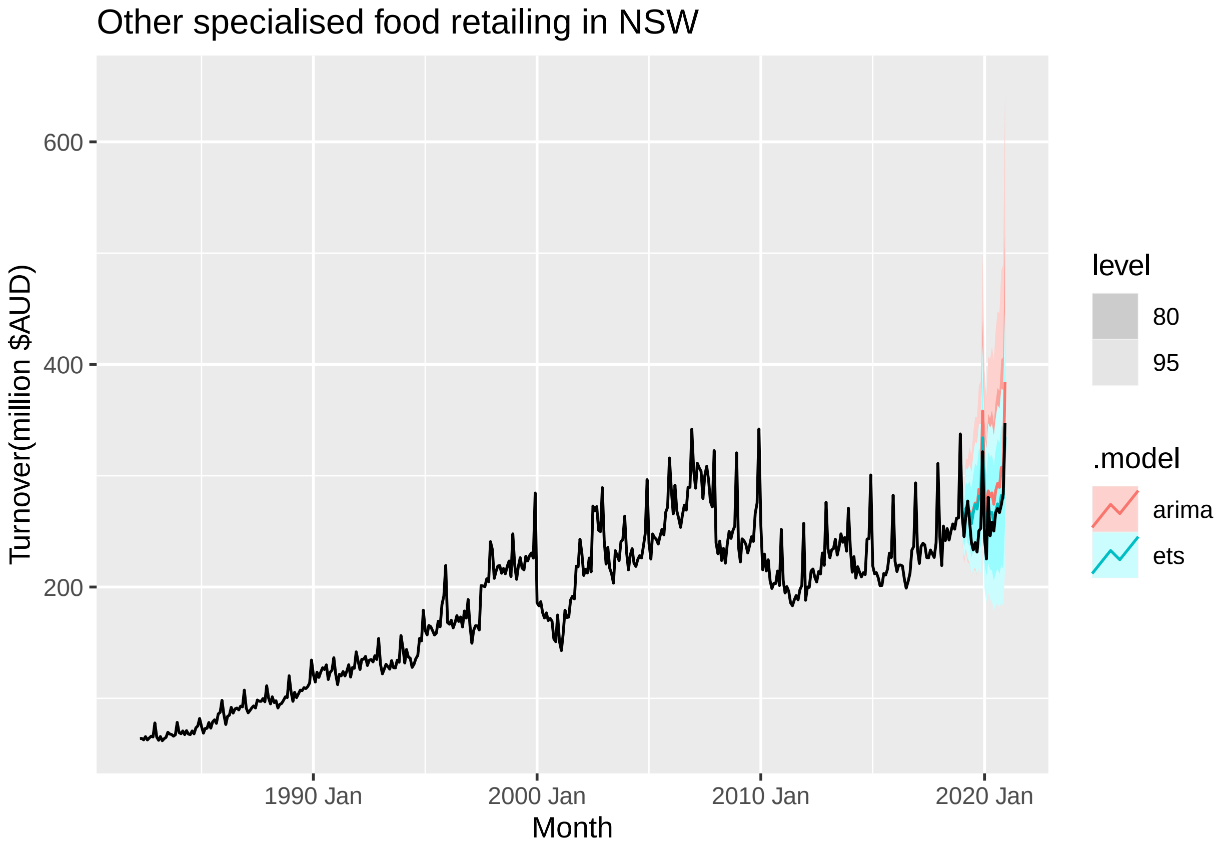
C. Comparing ETS and ARIMA

In forecasting the 2 years past the end of data, ETS model performs better in terms of accuracy, as it has lower RMSE and MASE, compared to the ARIMA model. This result is in contrast to the result obtained when we use the model to forecast last 24 months data, where ARIMA model has better accuracy.

```
model_combined_n <- myseries_n %>% model(ets=ETS(Turnover~error("M")+trend("Ad")+season("M")),
                                         arima=ARIMA(box_cox(Turnover,lambda_n)~0+pdq(0,1,2)+PDQ(0,1,2)
fc_combined_n <- model_combined_n %>% forecast(h="2 years")
fc_combined_n %>% accuracy(new_data) %>% arrange(RMSE)
```

```
## # A tibble: 2 x 12
##   .model State    Industry .type    ME  RMSE   MAE   MPE  MAPE  MASE RMSSE  ACF1
##   <chr>  <chr>    <chr>    <chr> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 ets    New Sou~ Other s~ Test  -12.1  19.0  15.2 -5.06  6.10 0.812 0.725 0.502
## 2 arima  New Sou~ Other s~ Test  -24.4  28.4  25.5 -9.51  9.90 1.36  1.09  0.559
```

```
model_combined_n %>% forecast(h="2 years") %>% autoplot(new_data) +labs(title="Other specialised food r
```



Benefit and Limitation of the Model

The advantage of ARIMA on this data is that we see a strong autocorrelation in the data. So, an ARIMA model can do well in incorporating past data to forecast into the future.

Some major disadvantages of ARIMA forecasting are the process of choosing the order of p,d,q and P,D,Q can be subjective. Thus, the reliability of the chosen model can depend on the skill and experience of the forecaster.It is required that the data to be fitted into the ARIMA model to be stationary, which might required us to do transformation and differecing (reflected in the order of d and D). However, there might be some limitation on how many differencing we could use, for example it would be better to use not more than 2 differencing. Also, ARIMA model cannot perform well in change of trend as well as the ETS model.

The benefit of ETS model is that it gives more weight to recent observations. It is also possible to adjust the parameter values to change how quickly older observation lose their importance. Furthermore, in this case, we have additional damped trend, which might performs well where the trend component is expected to be damped instead of being linear. ETS model also do not need the data to be stationary.