ETF3231/5231 Individual Assignment 4

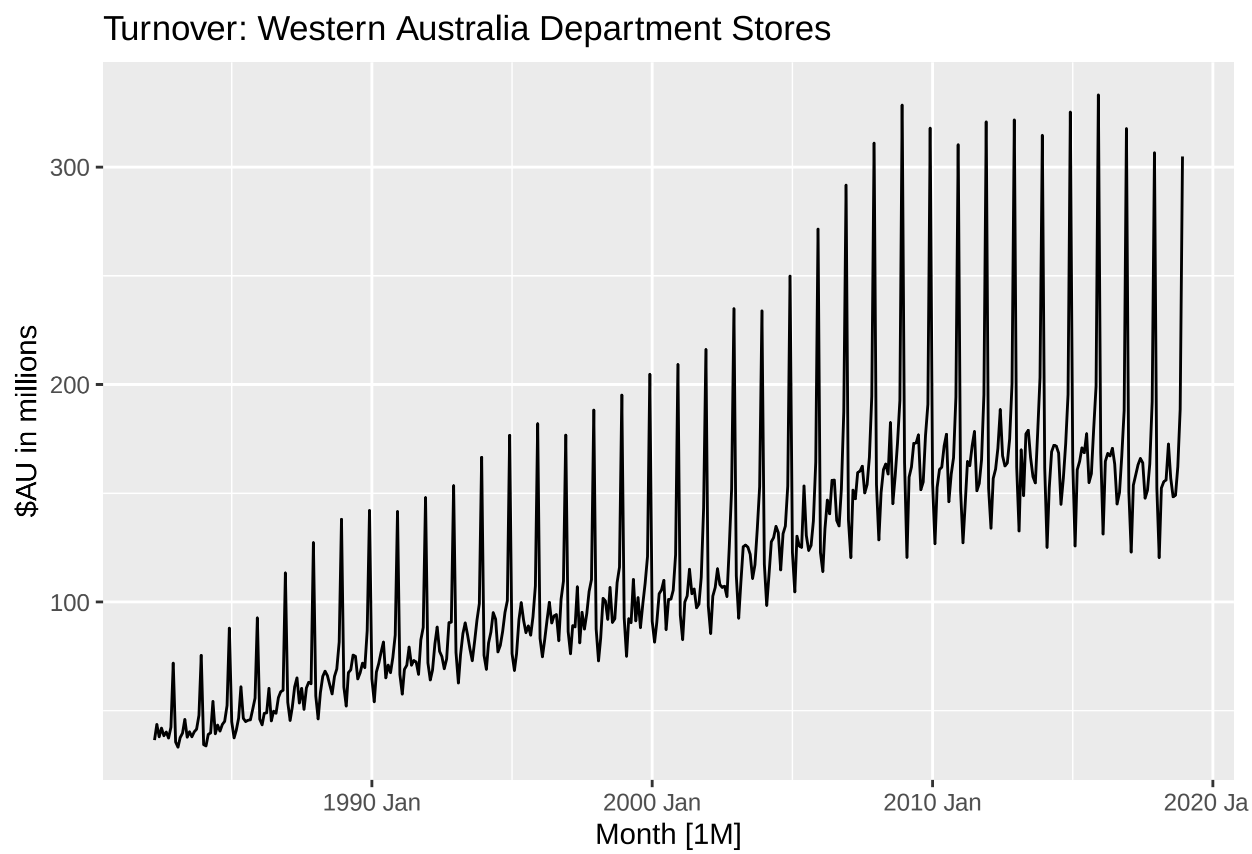
Kevin Tran 30631289

Your aim in the first part of the assignment is to build an ARIMA model and use this to forecast. Recall the first step in ARIMA modelling is to stabilize the variance of your data. If you decided that your data required a transformation in the previous assignments you will be required to use the same transformation for what follows (unless you have a reason to change your mind - please do so if you think it is necessary).

Question 1

Visually inspect your data and decide on the transformation and what differencing is required to achieve stationarity. Plot the data at every step and comment on each plot justifying your actions. (No more than 50 words per plot). (6 marks)

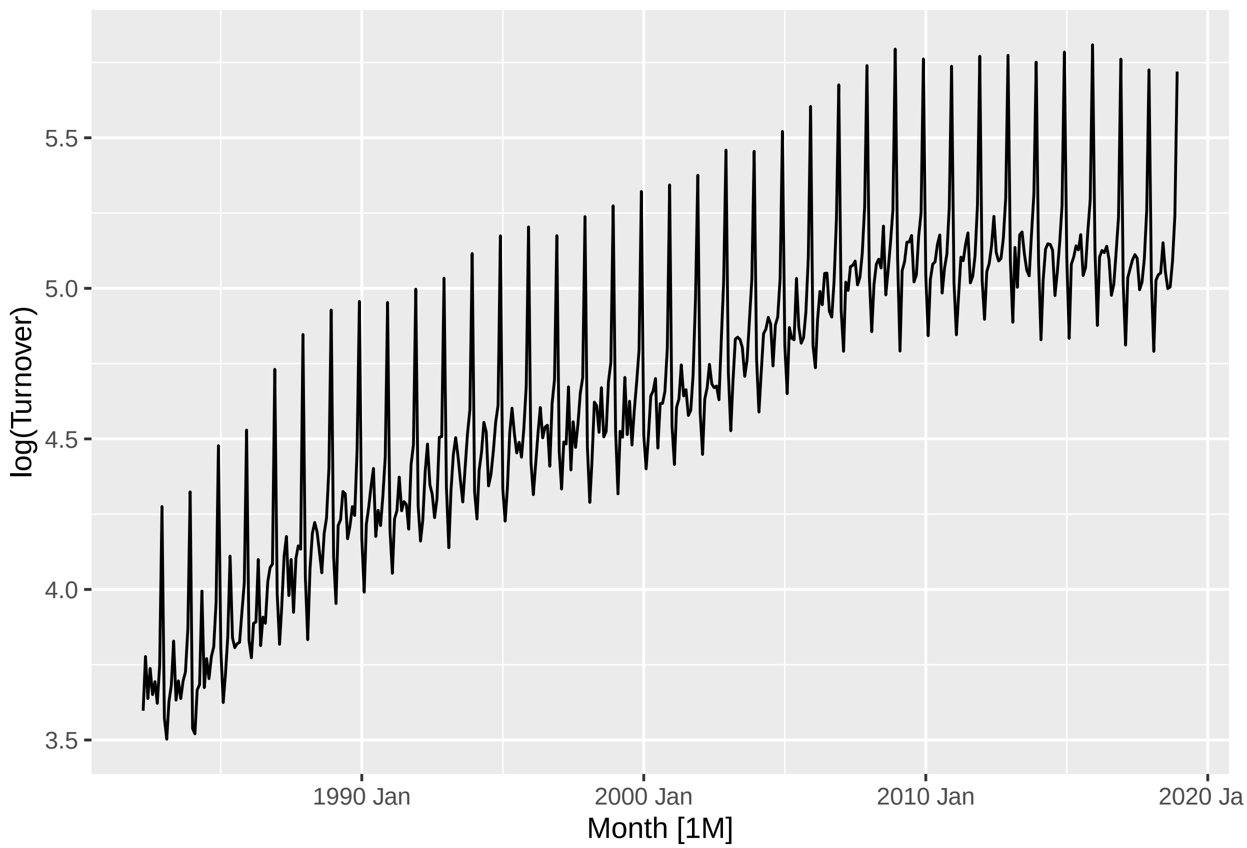
```
#Basic plot
my_series %>%
  autoplot(Turnover) +
  labs(
    title = "Turnover: Western Australia Department Stores ",
    y = "$AU in millions"
  )
```




```r
#Transformation
my_series %>%
  features(Turnover, features = guerrero)
```

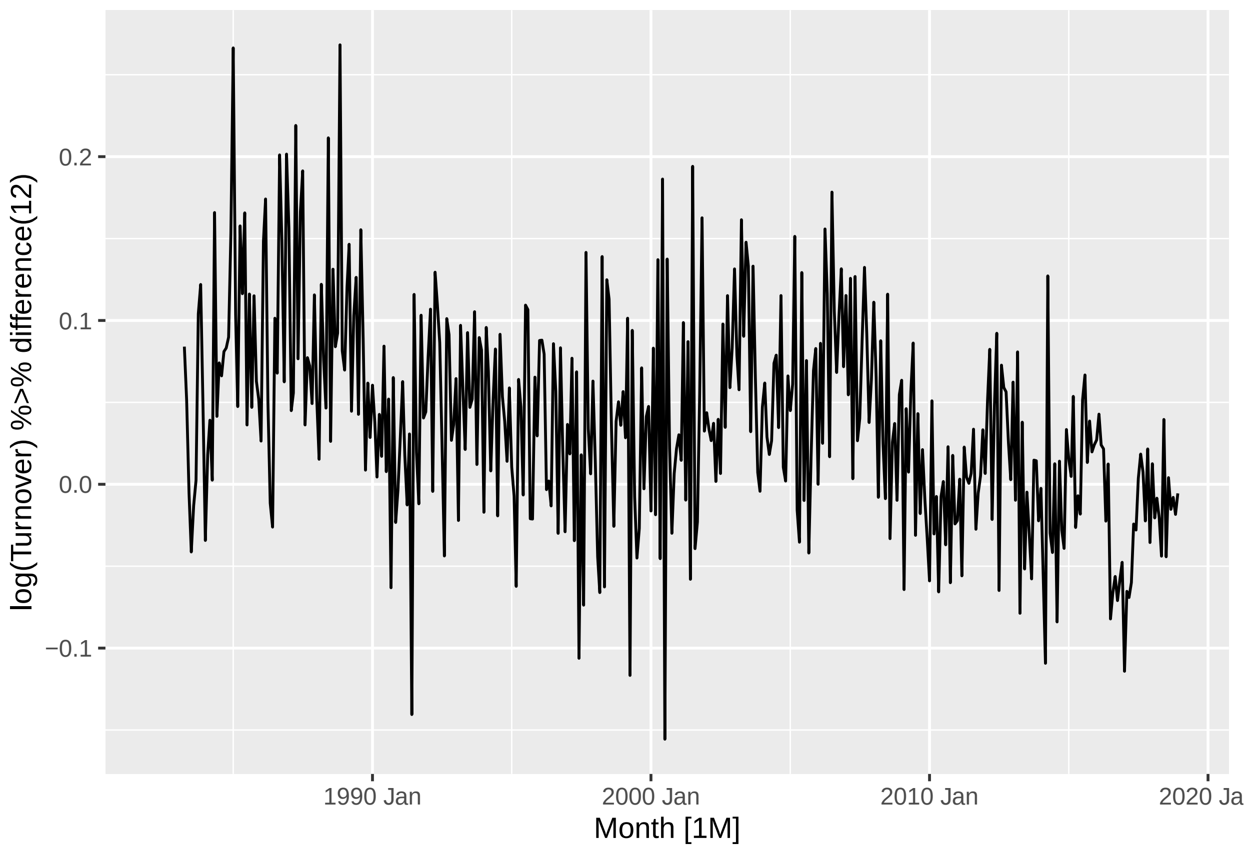
```
## # A tibble: 1 x 1
##   lambda_guerrero
##             <dbl>
## 1          0.0230
```

```r
my_series %>% autoplot(log(Turnover))
```




```
my_series %>% autoplot(log(Turnover)  %>%
  difference(12))
```

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

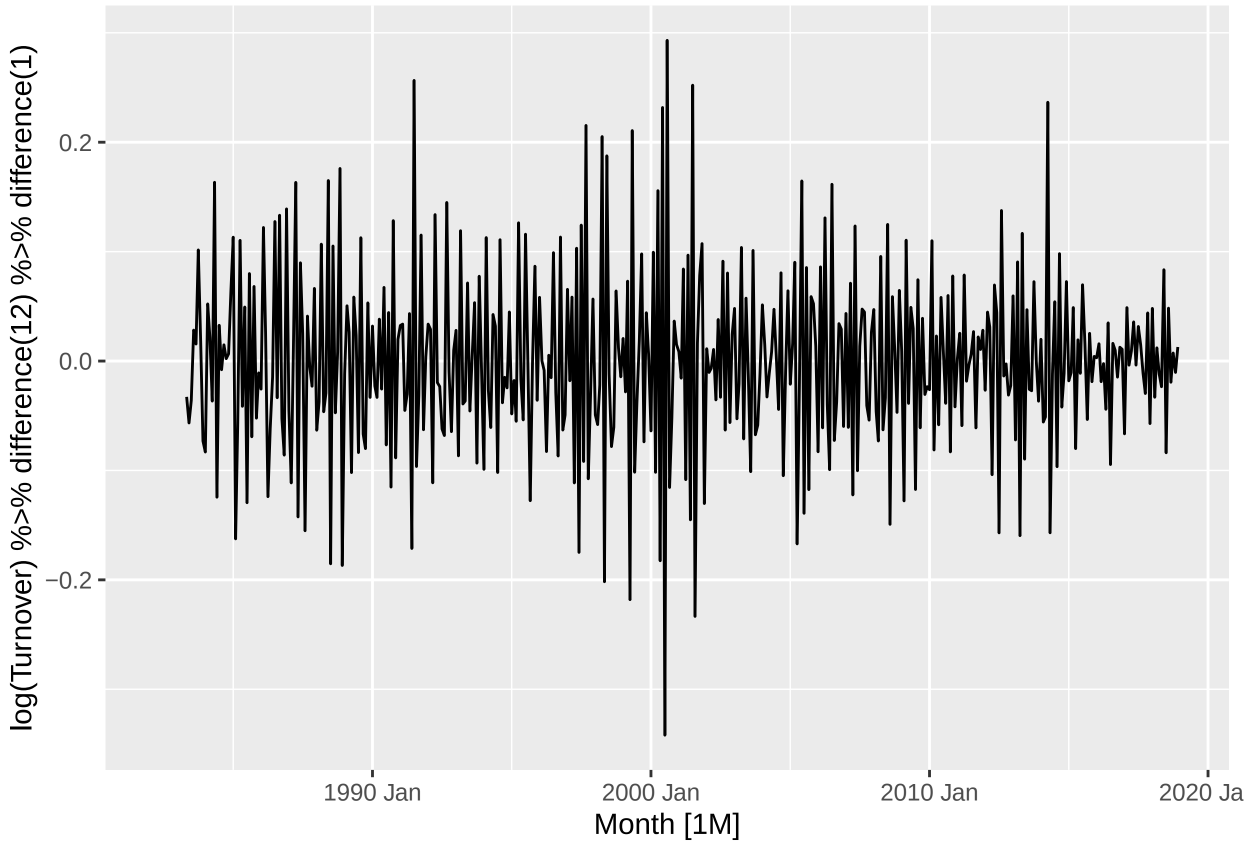



```r
#Automatic difference selection
my_series %>% mutate(log_turnover = log(Turnover)) %>%
  features(log_turnover, unitroot_nsdiffs)
```

```
## # A tibble: 1 x 1
##   nsdiffs
##     <int>
## 1       1
```

```r
my_series %>% autoplot(log(Turnover)  %>%
  difference(12) %>% difference (1))
```

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



The time series illustrates that the variance of the seasonality pattern increases. Therefore, a transformation is useful to stabilise the variance. Using the Box-Cox test, we get a value of 0.02. Hence a log transformation is more appropriate. However, stationarity is still not achieved as it shows an upward trend.

With the seasonal data, a seasonal difference of 12 is used first. But the time series is still not perfectly stationary as the variance is not constant. For instance, in the beginning of the time series the variance is larger than the variance at the end of the data.

Undergoing an automatic difference selection tells us first order differencing is required. Therefore, a difference of 1 is applied along with the seasonal difference to further enhance the stationary. Now the plot illustrates stationary clearly as it looks horizonal and now has a constant variance.

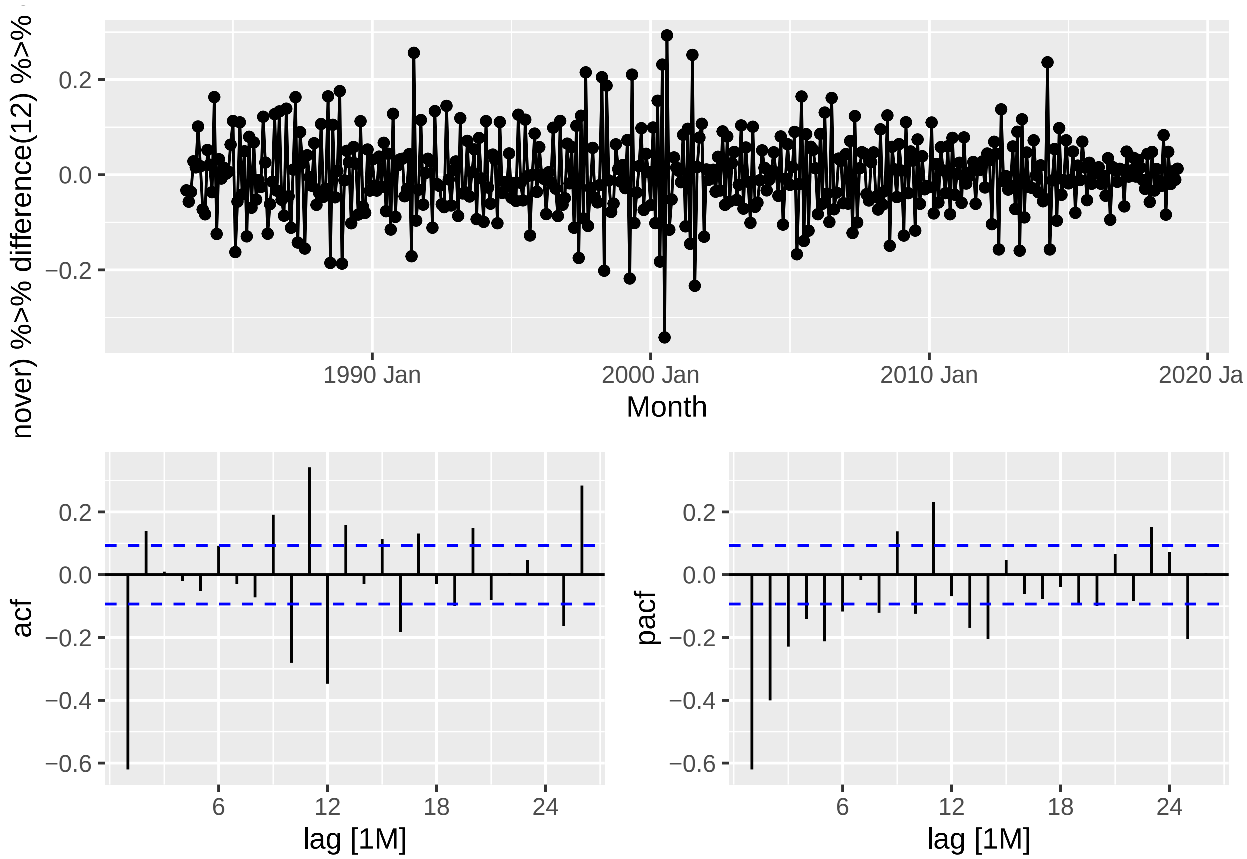
Question 2

Plot the ACF and PACF of the stationary data. Reading from these choose an appropriate ARIMA model. Make sure you justify your choice. (No more than 70 words in total – do not revise the theory – describe what you see in your plots and decide what ARIMA orders may be appropriate. Also note that it is highly likely that the ACF and PACF plots will be very messy. Do the best you can). (6 marks)

```
my_series %>%
  gg_tsdisplay(log(Turnover) %>% difference(12) %>% difference(1), plot_type = 'partial')
```

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

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



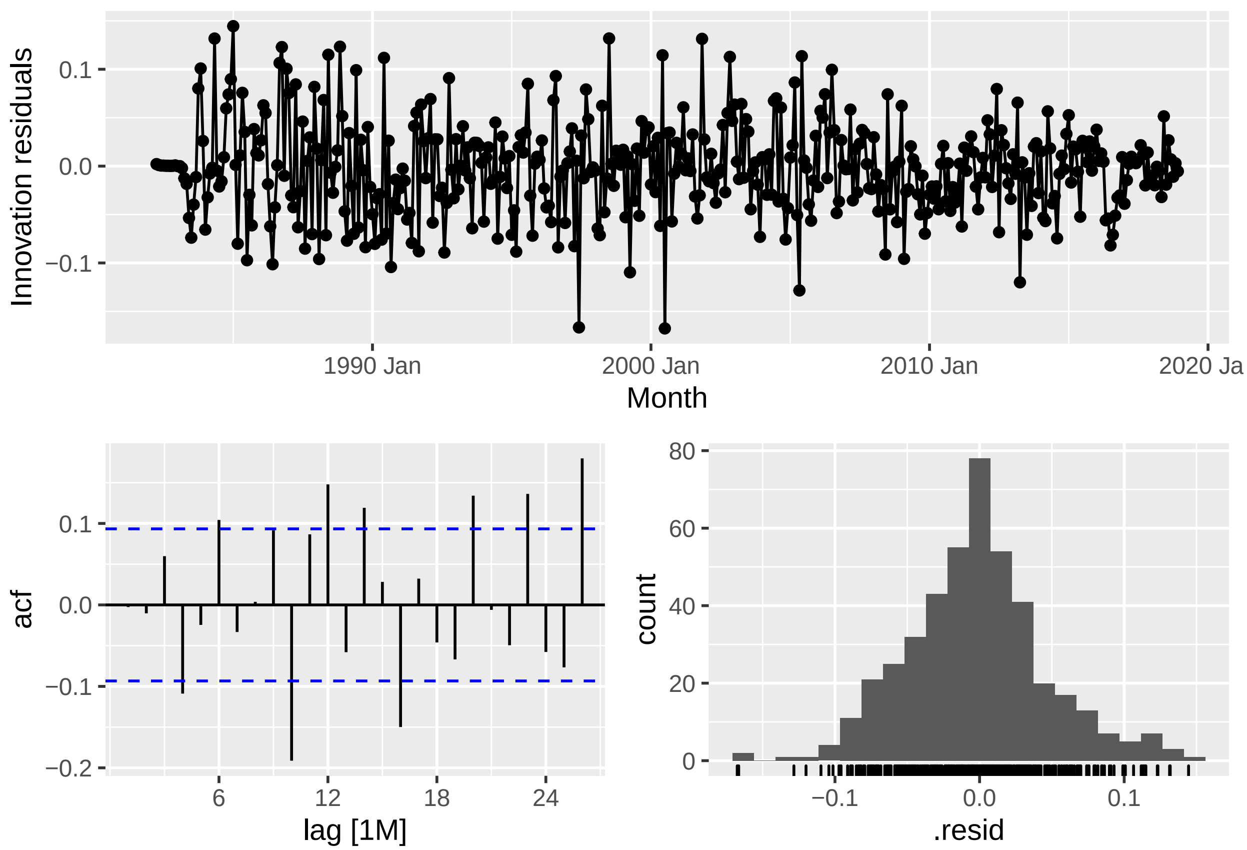
The data has been transformed into log format, with seasonality difference and first-order differencing. By first observing the ACF plot and the non-seasonal lags, there are 2 spikes at lags 11 and 26. Hence a MA(2) model is considered. For the seasonal lag, there is a spike at lag 12. Hence a MA(1) model is considered. Therefore, one possible ARIMA model is (0,1,2)(0,1,1) based on just the ACF plot.

Question 3

Check the whiteness of the residuals from the fitted ARIMA model. Based on these evaluate and if necessary review the ARIMA model specified in Q2. (No more than 50 words). (4 marks)

```
fit <- my_series %>%
  model(
    arima012011 = ARIMA(log(Turnover) ~ pdq(0,1,2) + PDQ(0,1,1))
  )

fit %>%
  select(arima012011) %>%
  gg_tsresiduals()
```

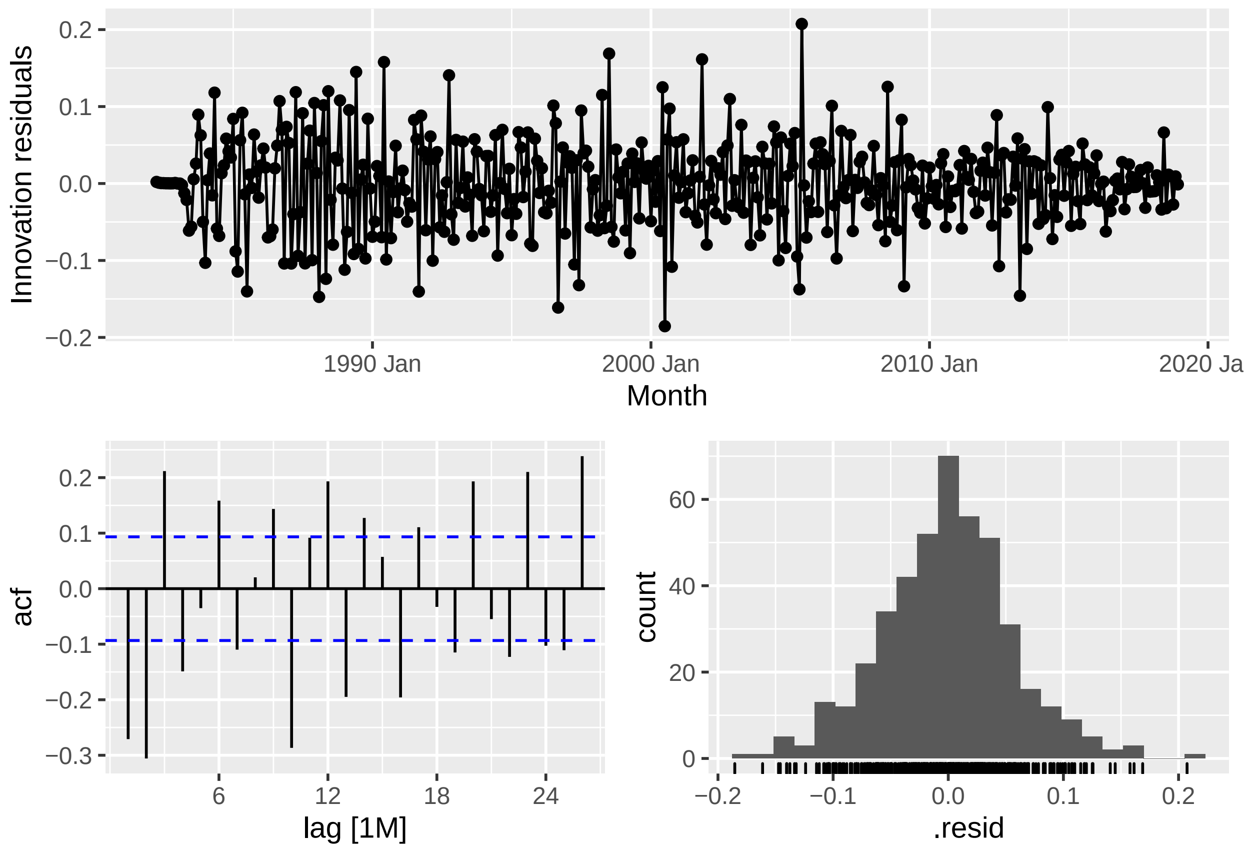


The residuals illustrate a heteroskedastic pattern. The histogram illustrates the distributions is normal with an outlier in the left tail. The ACF plot shows couple of small significant lags There is not a lot of white noise overall. Hence the ARIMA model could be a good fit.

Question 4

Consider three (up to five if you think you need them) alternative ARIMA models based on your choice in Q2 and Q3. (Very briefly justify each choice with no more than 1 or 2 lines each). Use information criteria to choose the best model you have considered so far. (6 marks)

```
fit_all <- my_series %>%
  model(
    arima012011 = ARIMA(log(Turnover) ~ pdq(0,1,2) + PDQ(0,1,1)),
    arima310011 = ARIMA(log(Turnover) ~ pdq(3,1,0) + PDQ(0,1,1)),
    arima110011 = ARIMA(log(Turnover) ~ pdq(1,1,0) + PDQ(0,1,1)),
    arima013011 = ARIMA(log(Turnover) ~ pdq(3,1,2) + PDQ(0,1,1))
  )
fit_all %>%
  select(arima110011) %>%
  gg_tsresiduals()
```




```
glance(fit_all)
```

```
## # A tibble: 4 x 8
##   .model       sigma2 log_lik    AIC   AICc    BIC ar_roots  ma_roots
##   <chr>         <dbl>   <dbl>  <dbl>  <dbl>  <dbl> <list>    <list>
## 1 arima012011 0.00240    680. -1352. -1352. -1336. <cpl [0]> <cpl [14]>
## 2 arima310011 0.00256    667. -1324. -1324. -1303. <cpl [3]> <cpl [12]>
## 3 arima110011 0.00332    610. -1214. -1214. -1202. <cpl [1]> <cpl [12]>
## 4 arima013011 0.00240    682. -1351. -1351. -1322. <cpl [3]> <cpl [14]>
```

The PACF plot in shows 3 spikes for non-seasonal lags so AR(3) can be considered. However the seasonal lags show exponential decay, therefore an MA(1) can be considered from the ACF plot that shows a significant spike at lag 12. Hence model (3,1,0)(0,1,1).

The PACF shows 1 spike at lag 11 for non-seasonal as the other two spike are quite similar, therefore AR(1) can be considered. Along with the seasonal lag from ACF plot (1,1,0)(0,1,1) can also be considered. There also seems to be less white noise and lot more spikes for this model after doing a residual diagnostic check.

The ACF plot of 2 spikes for non-seasonal lags and seasonal lag at spike 12, MA(2) and MA(1) can be considered. PACF shows 3 spikes for non-seasonal lags, AR(3) can be considered. Therefore the model (3,1,2)(0,1,1) can be considered.

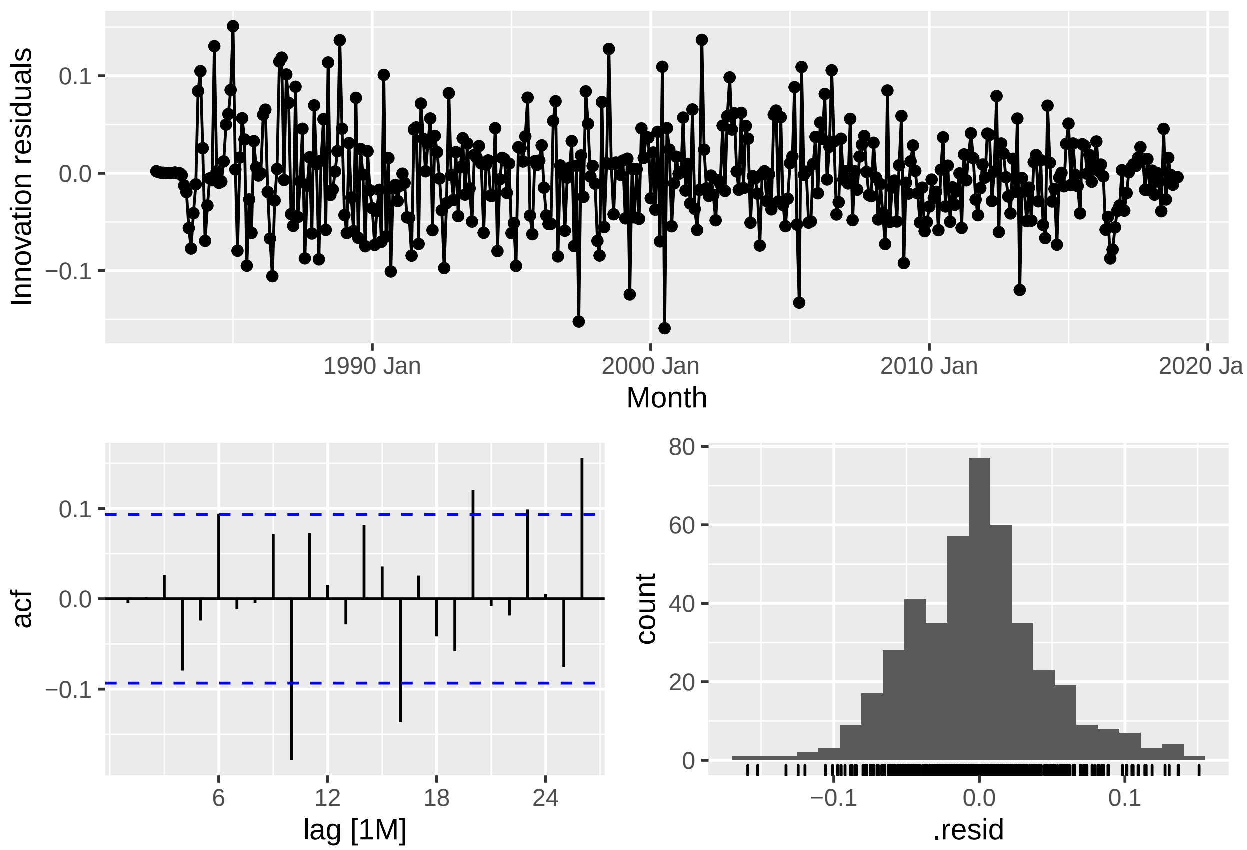
Despite the 3 models that have been considered, the ARIMA (0,1,2)(0,1,1) still outperforms the models mentioned above since it has the lowest AICc value.

Question 5

*Let the **ARIMA()** function choose a model. How does this compare with your chosen model from Q4? If you need to, make **ARIMA()** search harder exploring all possible options within the function. Perform a residual diagnostics analysis for your chosen model. (No more than 100 words). (6 marks)*

```
fit <- my_series %>%
  model(
    arima012011 = ARIMA(log(Turnover) ~ pdq(0,1,2) + PDQ(0,1,1)),
    fullsearch = ARIMA(log(Turnover), stepwise = FALSE, approx = FALSE)
  )

fit %>%
  select(fullsearch) %>%
  gg_tsresiduals()
```




```
glance(fit)
```

```
## # A tibble: 2 x 8
##   .model       sigma2 log_lik    AIC   AICc    BIC ar_roots    ma_roots
##   <chr>         <dbl>   <dbl>  <dbl>  <dbl>  <dbl> <list>      <list>
## 1 arima012011 0.00240    680. -1352. -1352. -1336. <cpl [0]>   <cpl [14]>
## 2 fullsearch  0.00232    688. -1366. -1366. -1346. <cpl [12]>  <cpl [14]>
```

```r
fit %>% select(fullsearch) %>% report()
```

```
## Series: Turnover
## Model: ARIMA(0,1,2)(1,1,1)[12]
## Transformation: log(Turnover)
##
## Coefficients:
##           ma1     ma2    sar1     sma1
##       -1.0359  0.2366  0.2582  -0.8324
## s.e.   0.0473  0.0464  0.0639   0.0387
##
## sigma^2 estimated as 0.002316:  log likelihood=687.99
## AIC=-1365.98   AICc=-1365.84   BIC=-1345.69
```

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

```
## # A tibble: 2 x 3
##   .model      lb_stat lb_pvalue
##   <chr>         <dbl>     <dbl>
## 1 arima012011    88.3  1.45e-10
## 2 fullsearch     53.5  6.91e- 5
```

The ARIMA function has selected the ARIMA (0,1,2)(1,1,1) model. The residuals show little difference, also illustrating a heteroskedastic pattern. As for the ACF plot, it shows a significant spike at lag 10, followed by a couple of small spikes lags 16, 20 and 26. The histogram also shows a normal distribution with no outliers at either end of tails. If we were to perform a white-noise test, the p-value is small, indicating residuals are correlated.

Question 6

Remember that you cannot use information criteria to compare between models with different orders of differencing. **If necessary** *use an appropriate test set to choose the ARIMA model you want to use for forecasting. Which model have you selected and why? (No more than 50 words). (4 marks)*

```
fit_all <- my_series %>%
  model(
    arima012011 = ARIMA(log(Turnover) ~ pdq(0,1,2) + PDQ(0,1,1)),
    arima310011 = ARIMA(log(Turnover) ~ pdq(3,1,0) + PDQ(0,1,1)),
    arima110011 = ARIMA(log(Turnover) ~ pdq(1,1,0) + PDQ(0,1,1)),
    arima013011 = ARIMA(log(Turnover) ~ pdq(3,1,2) + PDQ(0,1,1)),
    fullsearch = ARIMA(log(Turnover), stepwise = FALSE, approx = FALSE)
  )
glance(fit_all)
```

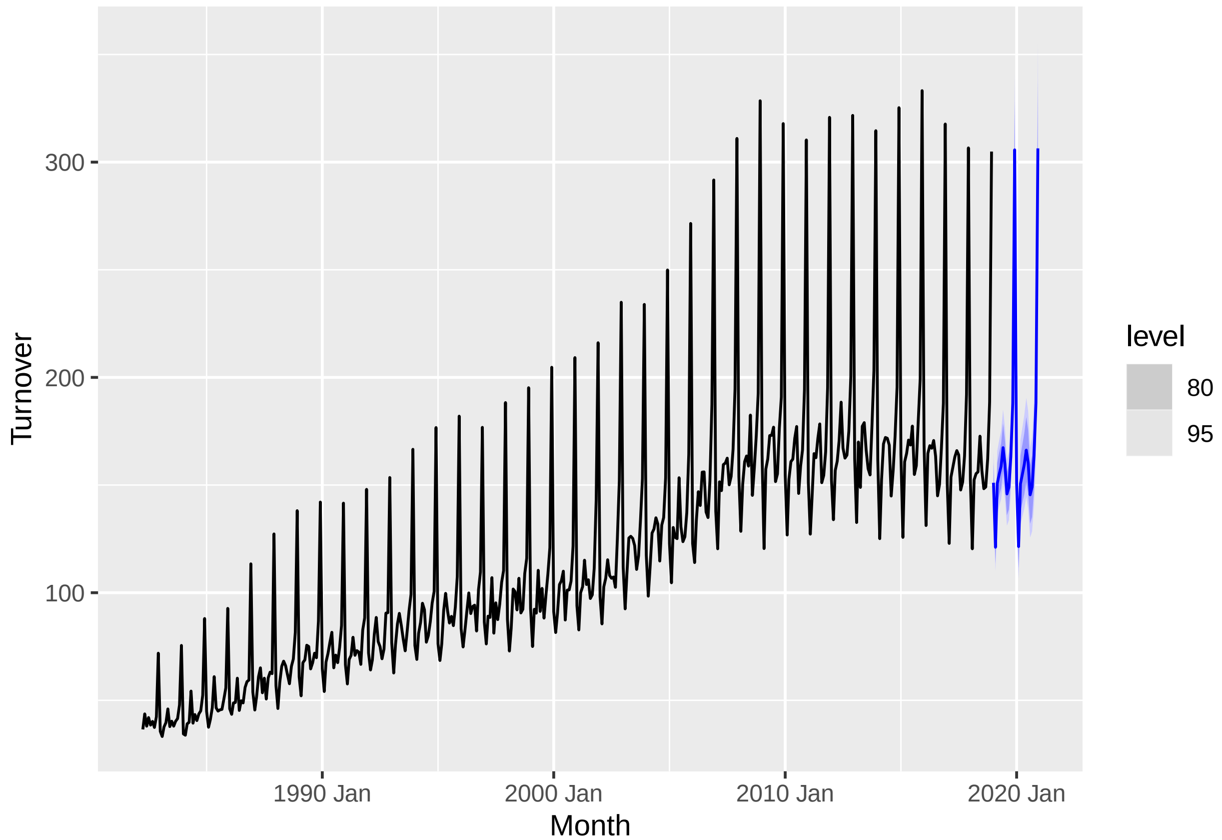
```
## # A tibble: 5 x 8
##   .model       sigma2 log_lik    AIC   AICc    BIC ar_roots    ma_roots
##   <chr>         <dbl>   <dbl>  <dbl>  <dbl>  <dbl> <list>      <list>
## 1 arima012011 0.00240    680. -1352. -1352. -1336. <cpl [0]>  <cpl [14]>
## 2 arima310011 0.00256    667. -1324. -1324. -1303. <cpl [3]>  <cpl [12]>
## 3 arima110011 0.00332    610. -1214. -1214. -1202. <cpl [1]>  <cpl [12]>
## 4 arima013011 0.00240    682. -1351. -1351. -1322. <cpl [3]>  <cpl [14]>
## 5 fullsearch  0.00232    688. -1366. -1366. -1346. <cpl [12]> <cpl [14]>
```

The models considered this far have all used seasonal and first-order differencing. Hence AICc is appropriate for comparison. Observing the AICc, we find that that the ARIMA(0,1,2)(1,1,1) model has the lowest value. Therefore, the model that R has chosen will be used for forecasting.

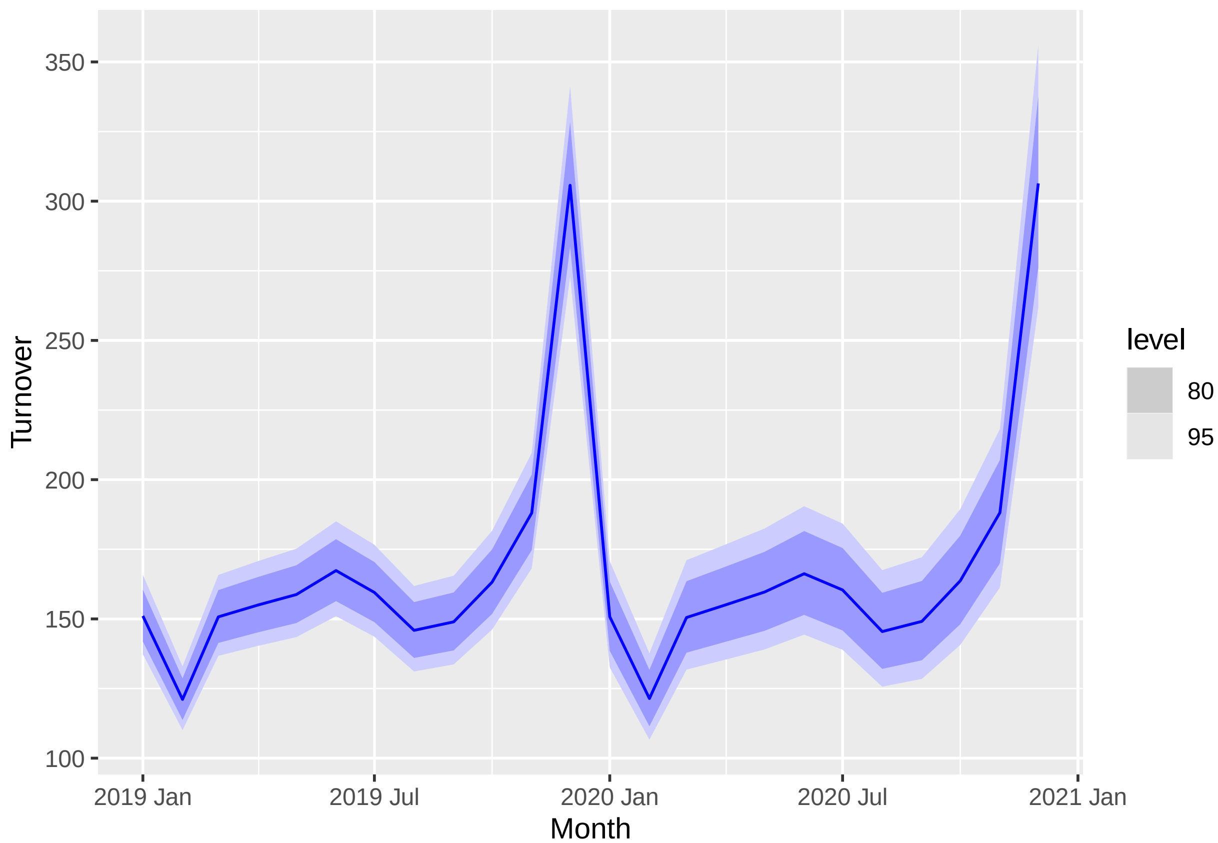
Question 7

Generate and plot forecasts and forecast intervals from your chosen ARIMA model for two years following the end of your sample. Comment on these. (No more than 50 words). (3 marks)

```
fit %>%
  select(fullsearch) %>%
  forecast(h = "2 years") %>%
  autoplot(my_series)
```




```
fit %>%
  select(fullsearch) %>%
  forecast(h = "2 years") %>%
  autoplot()
```



The ARIMA (0,1,2)(1,1,1) model forecast captured the seasonality quite well as the pattern almost resembles with the original data. In addition, the forest intervals do not appear too wide where the seasonality peaks occur but quite wide in between.

Question 8

You have now considered several modelling frameworks and built several models for your data set. In this part of the assignment you will evaluate these.

Find your data on the ABS website and update your series till the end of the currently available data. Explore your updated time series and comment on the effects of the COVID-19 pandemic. Provide any necessary plots to support your analysis. Some States and Industries are unfortunately affected more than others. (6 marks)

```
#updated data
abs_data <- readabs::read_abs(series_id="A3349434X") %>%
  select(date,value) %>%
  mutate(date=yearmonth(date)) %>%
  as_tsibble(index = date)
```

```
## Finding URLs for tables corresponding to ABS series ID
```

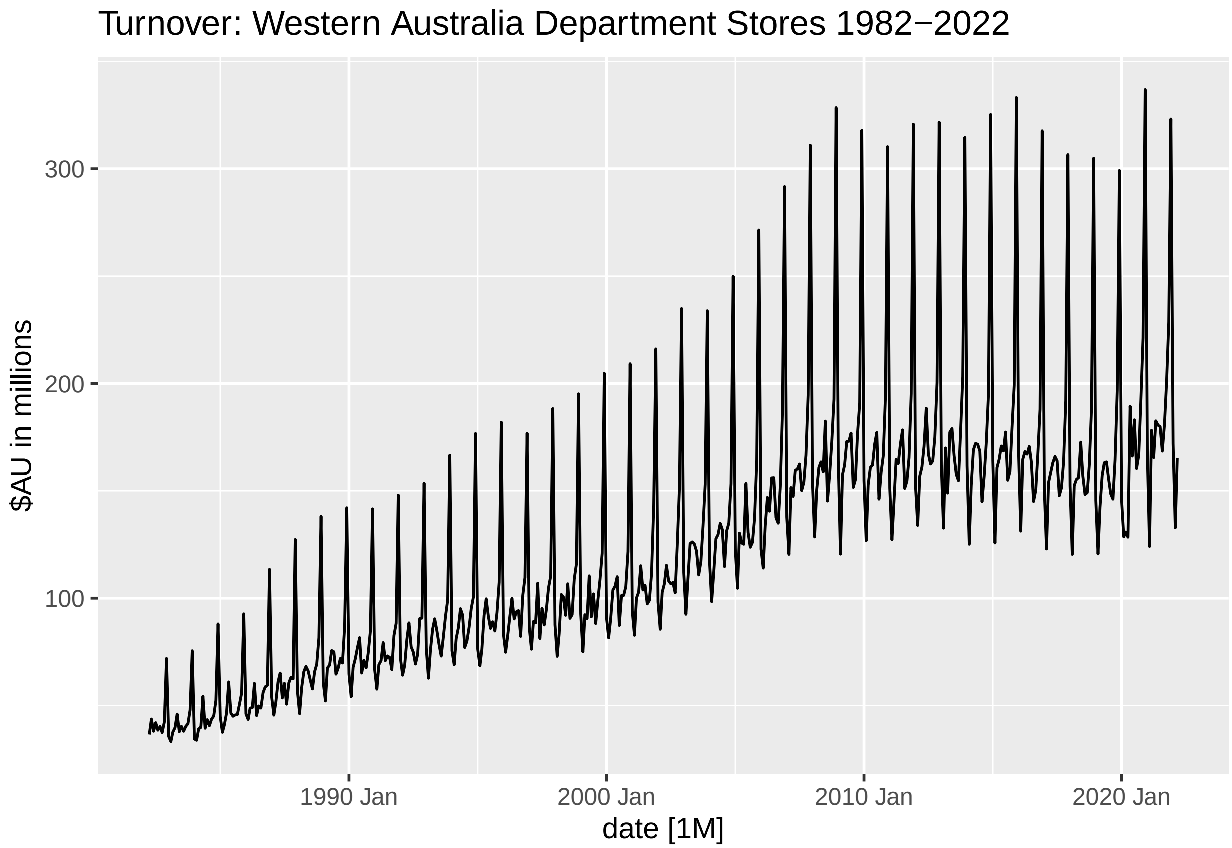
```
## Attempting to download files from series ID , Retail Trade, Australia
```

```
## Downloading https://www.abs.gov.au/statistics/industry/retail-and-wholesale-trade/retail-trade-austra
```

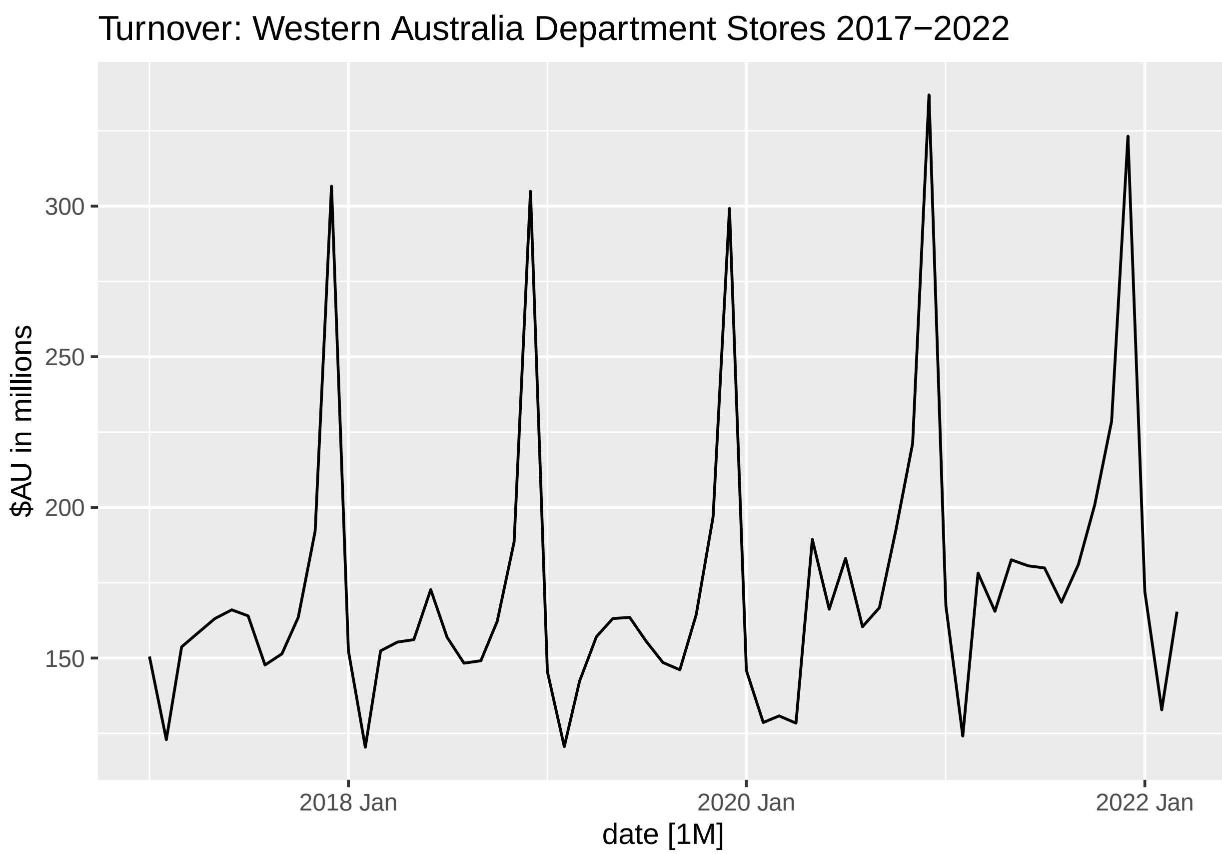
```
## Extracting data from downloaded spreadsheets
```

```
## Tidying data from imported ABS spreadsheets
```

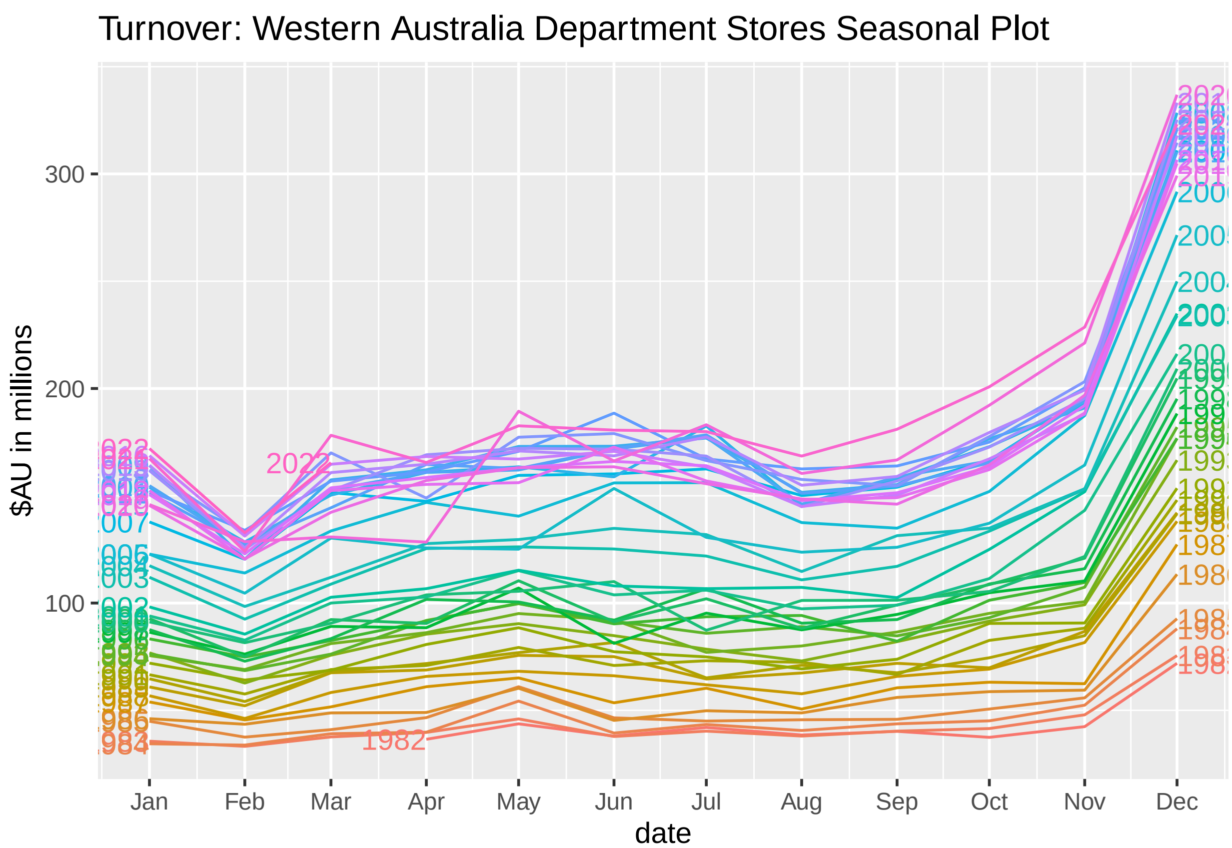
```
abs_data %>% autoplot(value) +   labs(
    title = "Turnover: Western Australia Department Stores 1982-2022 ",
    y = "$AU in millions"
  )
```



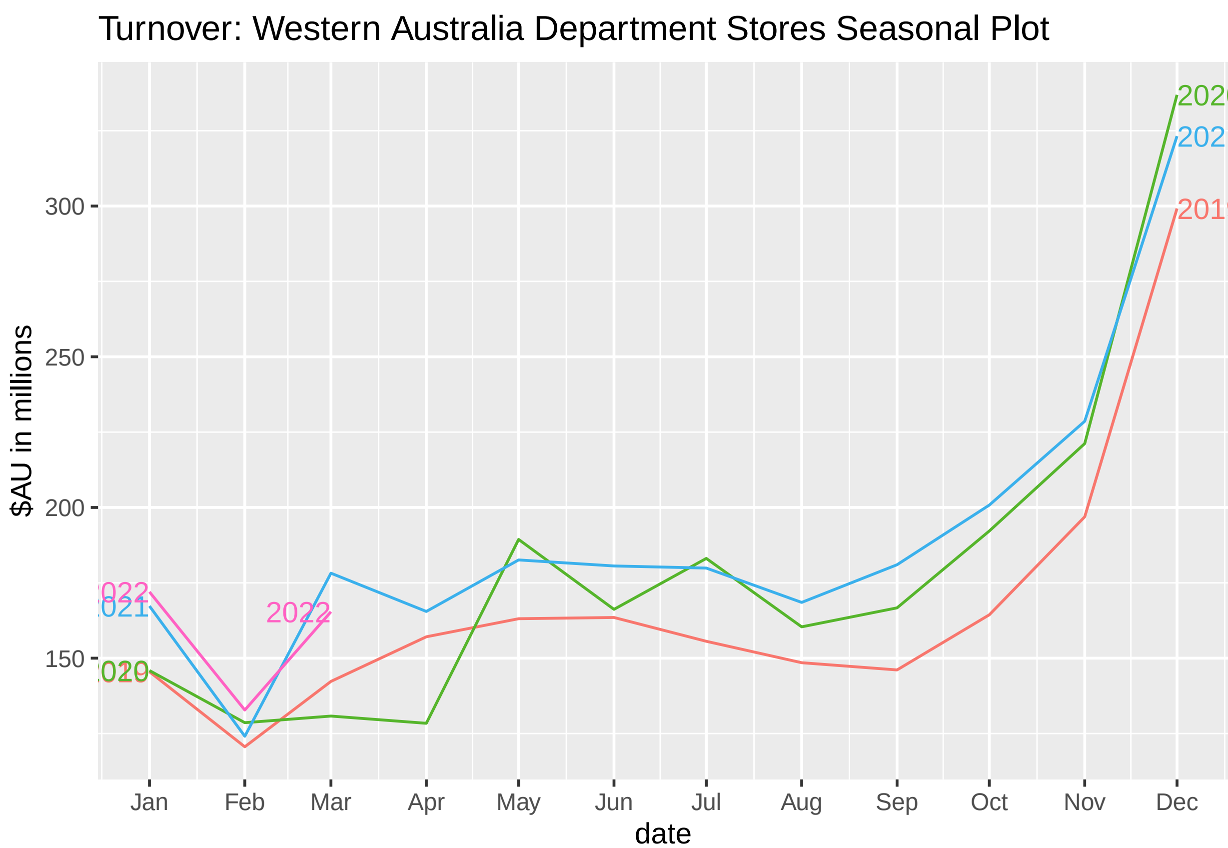

```r
#zoomed in
abs_data %>%
  filter(year(date)>=2017) %>%
  autoplot(value) +   labs(
    title = "Turnover: Western Australia Department Stores 2017-2022 ",
    y = "$AU in millions"
  )
```



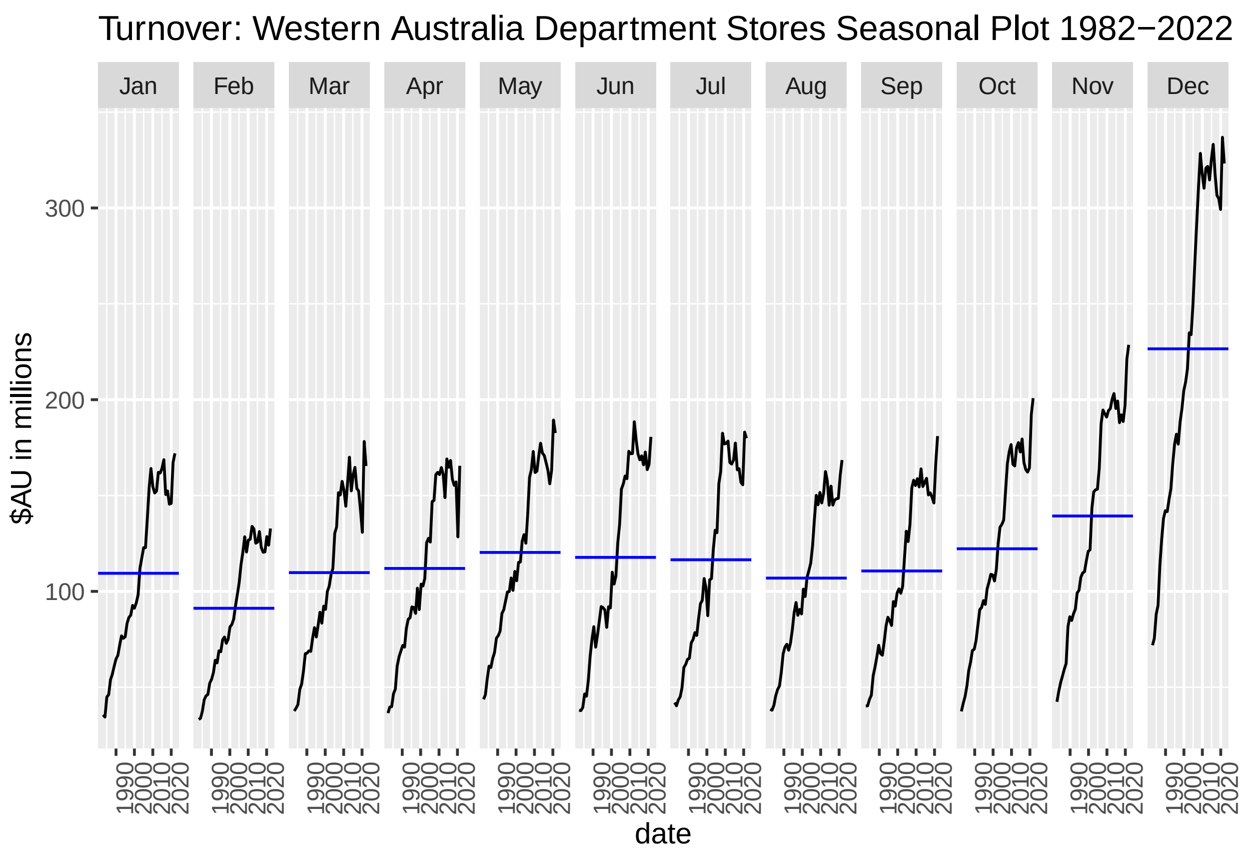

```
#seasonal plot
abs_data %>%
  gg_season(value, labels = 'both') + guides(colour = "none") +
  labs(
    title = "Turnover: Western Australia Department Stores Seasonal Plot ",
    y = "$AU in millions"
  )
```



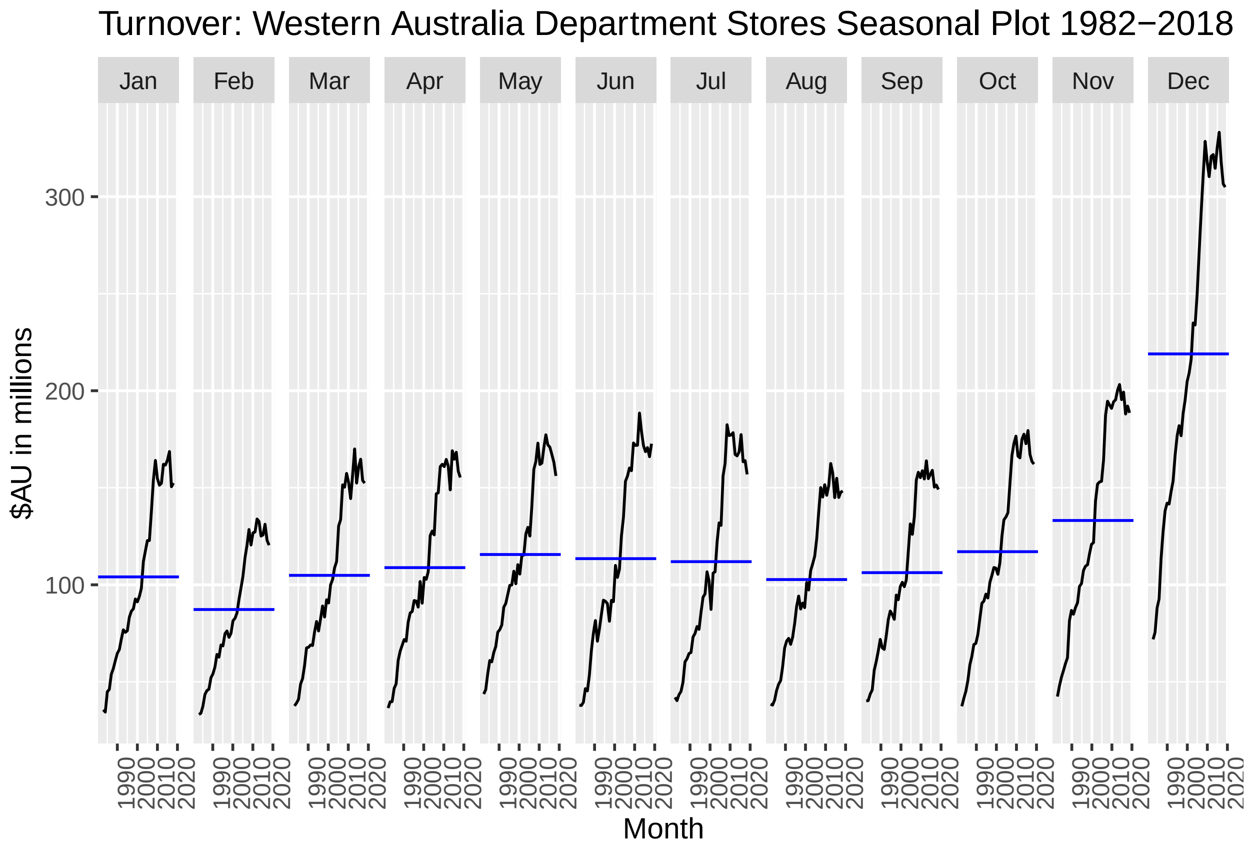

```
abs_data %>%
  filter(year(date)>2018) %>%
  gg_season(value, labels = 'both') + guides(colour = "none") +
  labs(
    title = "Turnover: Western Australia Department Stores Seasonal Plot ",
    y = "$AU in millions"
  )
```




```
#subseries plot
abs_data %>%
  gg_subseries(value) +
  labs(
    title = "Turnover: Western Australia Department Stores Seasonal Plot 1982-2022 ",
    y = "$AU in millions"
  )
```




```r
my_series %>%
  gg_subseries(Turnover) +
  labs(
    title = "Turnover: Western Australia Department Stores Seasonal Plot 1982-2018 ",
    y = "$AU in millions"
  )
```



The updated data now contains turnovers for department stores in Western Australia from April 1982 to March 2022. The time series still illustrates an upward trend with seasonality patterns peaking every end of the year. However, from 2020 onwards, there seems to be some cyclic patterns between right before a seasonality peak.

To analyse this further, the seasonal plot for 2020 and 2021 show that these cyclic peaks are occurring around the first half of the year (around April to July). The peak from April to May in 2021 could be capturing when Western Australia first went into lockdown near the end of March and many citizens went into panic buying. Furthermore, the time series for 2020 and 2021 seem to be higher than 2019. A possible reason could be that many department stores offering online services have caused the turnovers to increase.

To demonstrate, the subseries plot for the updated data has also showed that the mean of turnover's has jumped slightly across all 12 months. It seems that the COVID-19 pandemic did not dramatically affect Western Australia department stores.

Question 9

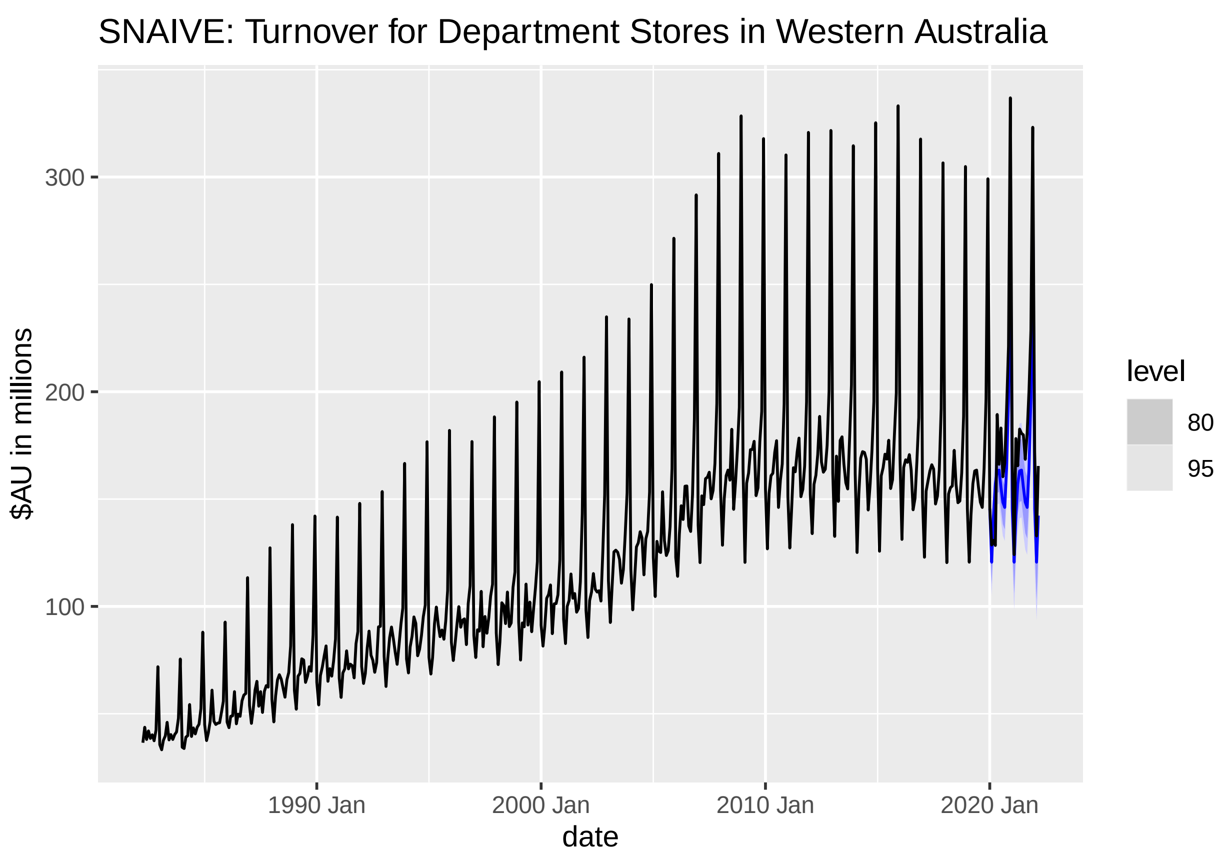
Generate forecasts for the period post 2019 until the end of your sample, from the models considered "best" in all assignments. More specifically, generate forecasts from the best benchmark, the best ETS and best ARIMA model. Plot the forecasts (both point forecasts and prediction intervals) together with the observed data and comment on these. (Make sure you can clearly visualise these. You may choose to plot on multiple graphs.) (6 marks)

```
#training 1982 - 2019
abs_data_train <- abs_data %>% slice(1:453)
#test 2020-2022
abs_data_test <- abs_data %>%
  filter(year(date) >= 2020)
```

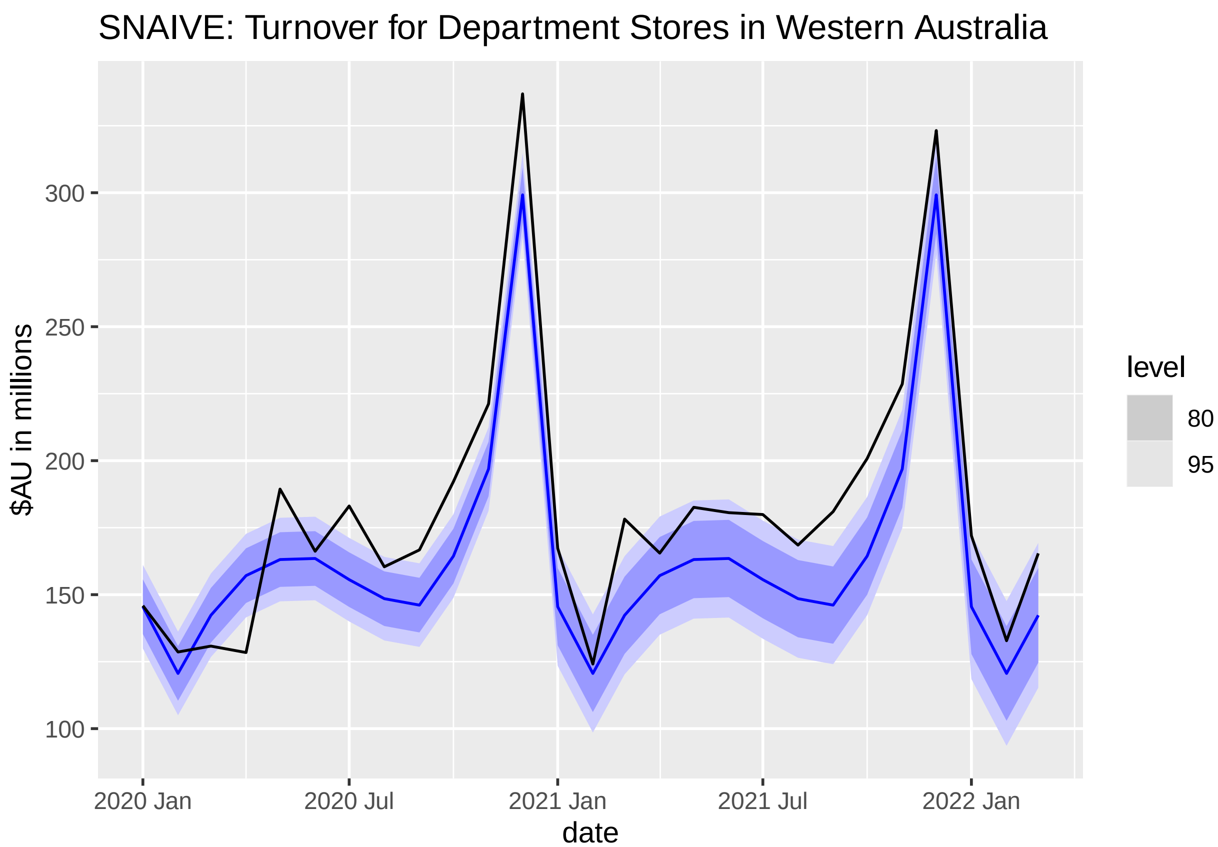
```
#benchmark -  seasonal naive model was chosen best in previous. So hence u this.
abs_data_benchmark_fit <-  abs_data_train %>%
  model(Seasonal_naive = SNAIVE(value))

#forecast 27 months from dec 2019 since our data ends on march 2022.
abs_data_benchmark_fc <- abs_data_benchmark_fit %>%
  forecast(h = '27 months')

abs_data_benchmark_fc %>% autoplot(abs_data) +
  labs(title = "SNAIVE: Turnover for Department Stores in Western Australia",
       y = "$AU in millions") +
  guides(colour = guide_legend(title = "Forecast"))
```



```r
abs_data_benchmark_fc %>% autoplot(abs_data_test) +
  labs(title = "SNAIVE: Turnover for Department Stores in Western Australia",
       y = "$AU in millions") +
  guides(colour = guide_legend(title = "Forecast"))
```



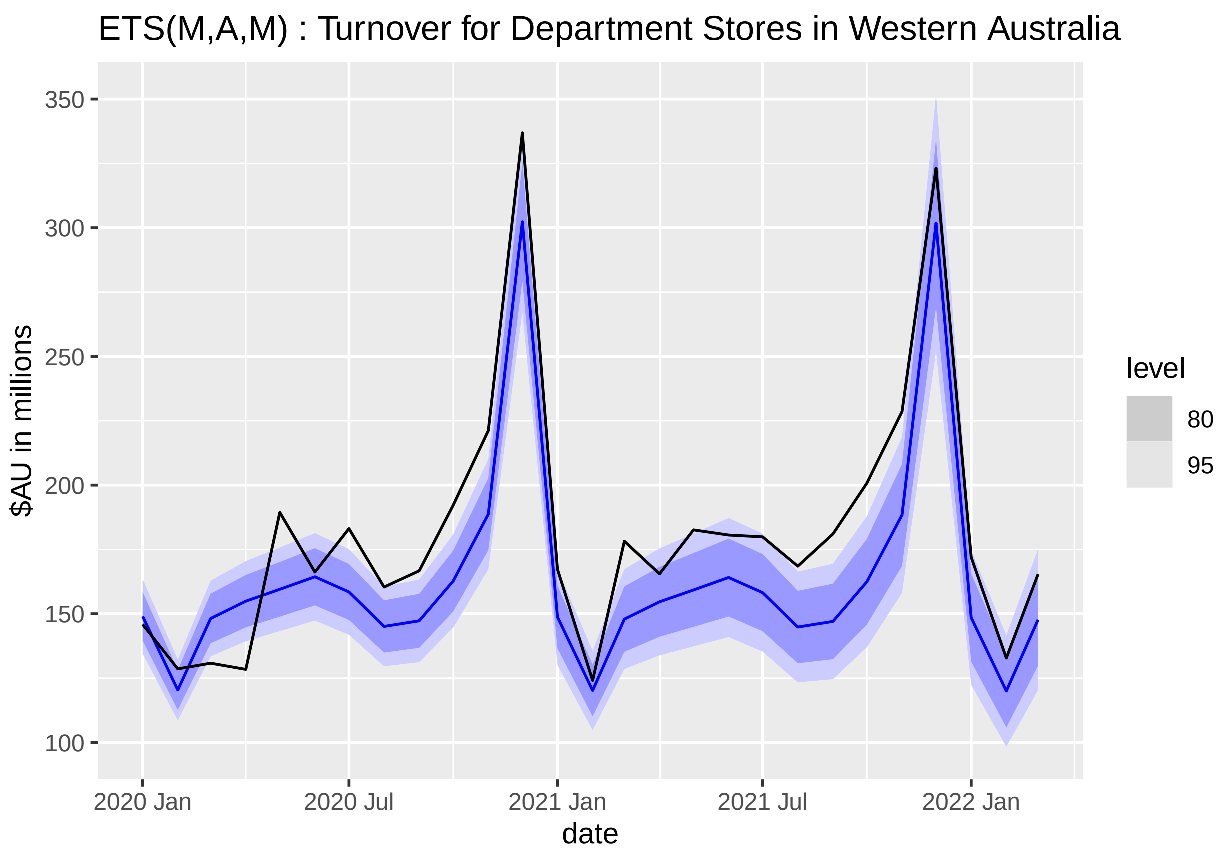

```r
benchmark_point_intervals <- abs_data_train%>%
  model(SNAIVE(value)) %>%
  forecast(h = '27 months') %>%
  hilo()
print(benchmark_point_intervals)
```

```
## # A tsibble: 27 x 6 [1M]
## # Key:       .model [1]
##    .model            date       value .mean                  `80%`
##    <chr>            <mth>      <dist> <dbl>                 <hilo>
##  1 SNAIVE(value) 2020 Jan N(146, 63)  146. [135.3062, 155.6938]80
##  2 SNAIVE(value) 2020 Feb N(121, 63)  121. [110.4062, 130.7938]80
##  3 SNAIVE(value) 2020 Mar N(142, 63)  142. [132.1062, 152.4938]80
##  4 SNAIVE(value) 2020 Apr N(157, 63)  157. [146.9062, 167.2938]80
##  5 SNAIVE(value) 2020 May N(163, 63)  163. [152.9062, 173.2938]80
##  6 SNAIVE(value) 2020 Jun N(164, 63)  164. [153.3062, 173.6938]80
##  7 SNAIVE(value) 2020 Jul N(156, 63)  156. [145.4062, 165.7938]80
##  8 SNAIVE(value) 2020 Aug N(148, 63)  148. [138.3062, 158.6938]80
##  9 SNAIVE(value) 2020 Sep N(146, 63)  146. [135.9062, 156.2938]80
## 10 SNAIVE(value) 2020 Oct N(164, 63)  164. [154.2062, 174.5938]80
```

```
## # ... with 17 more rows, and 1 more variable: ‘95%‘ <hilo>
```

```
#ETS-  MAM chosen as best model in previous.
ETS_fit <- abs_data_train %>%
  model(
    hw_multi = ETS(value ~ error("M") + trend("A") +
                                                season("M")))
ETS_fc <- ETS_fit %>%
  forecast(h = '27 months')

ETS_fc  %>% autoplot(filter(abs_data, year(date)>2019)) +
  labs(title = "ETS(M,A,M) : Turnover for Department Stores in Western Australia",
       y = "$AU in millions") +
  guides(colour = guide_legend(title = "Forecast"))
```



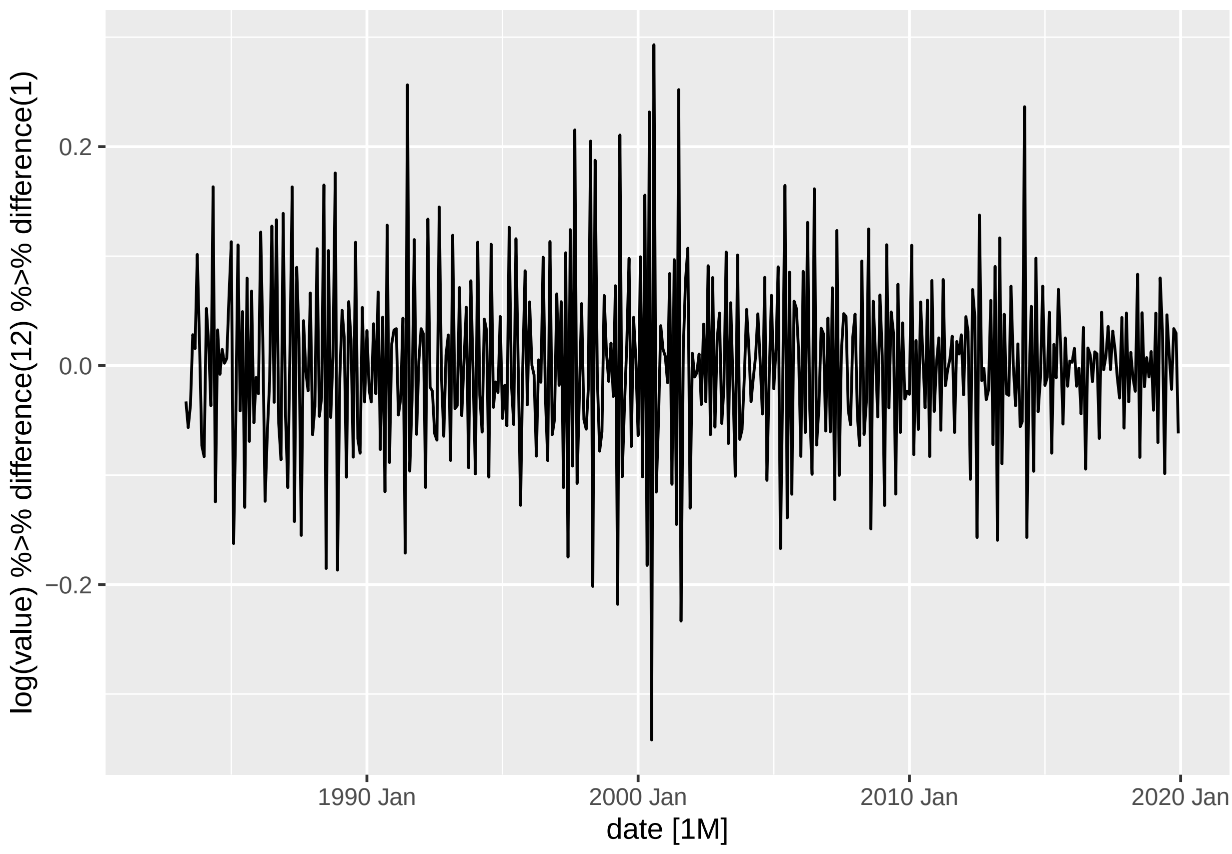

```
ETS_point_intervals <- ETS_fit %>%
  forecast(h= '27  months') %>%
  hilo()
print(ETS_point_intervals)
```

```
## # A tsibble: 27 x 6 [1M]
## # Key:       .model [1]
##    .model       date      value .mean                   ‘80%‘
##    <chr>       <mth>     <dist> <dbl>                  <hilo>
##  1 hw_multi 2020 Jan N(149, 55)  149. [139.5260, 158.4840]80
```

```
##  2 hw_multi 2020 Feb N(120, 36)  120. [112.6626, 128.1208]80
##  3 hw_multi 2020 Mar N(148, 56)  148. [138.5560, 157.7826]80
##  4 hw_multi 2020 Apr N(155, 63)  155. [144.7288, 165.0709]80
##  5 hw_multi 2020 May N(160, 69)  160. [148.8972, 170.1282]80
##  6 hw_multi 2020 Jun N(164, 75)  164. [153.2730, 175.4791]80
##  7 hw_multi 2020 Jul N(158, 72)  158. [147.5966, 169.3575]80
##  8 hw_multi 2020 Aug N(145, 63)  145. [134.9513, 155.2293]80
##  9 hw_multi 2020 Sep N(147, 67)  147. [136.7814, 157.7591]80
## 10 hw_multi 2020 Oct N(163, 85)  163. [150.9080, 174.5634]80
## # ... with 17 more rows, and 1 more variable: '95%' <hilo>
```

```
#ARIMA - best arima model ARIMA(0,1,2)(1,1,1)
abs_data_train %>% autoplot(log(value)  %>%
  difference(12) %>% difference (1))
```

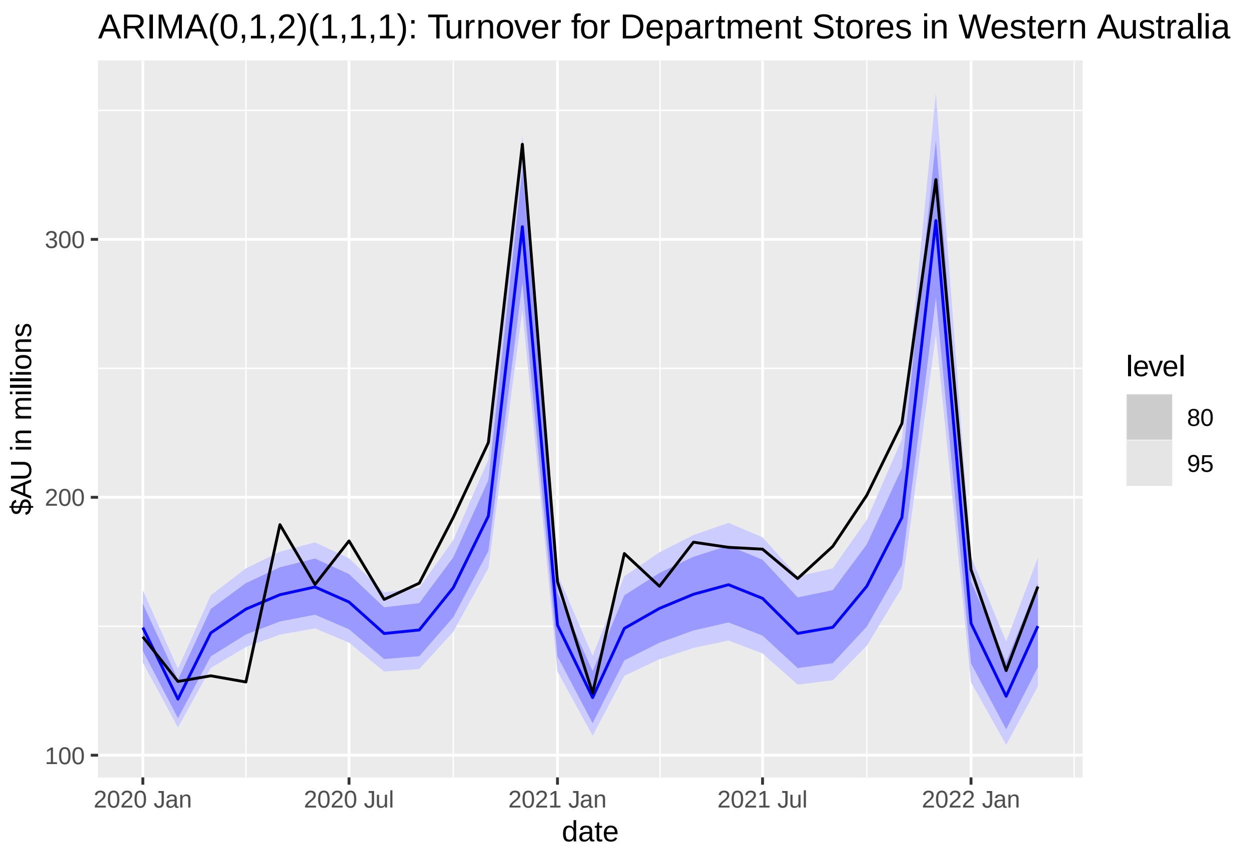
```
## Warning: Removed 13 row(s) containing missing values (geom_path).
```




```
arima_fit <- abs_data_train %>%
  model(
    arima012111 = ARIMA(log(value) ~ pdq(0,1,2) + PDQ(1,1,1))
  )

arima_fc <- arima_fit %>%
  select(arima012111) %>%
  forecast(h = "27 months")
```

```r
arima_fc %>% autoplot(filter(abs_data, year(date)>2019)) +
   labs(title = "ARIMA(0,1,2)(1,1,1): Turnover for Department Stores in Western Australia",
        y = "$AU in millions") +
  guides(colour = guide_legend(title = "Forecast"))
```




```r
arima_point_intervals <- arima_fit %>%
  forecast(h= '27 months') %>%
  hilo()
arima_point_intervals
```

```
## # A tsibble: 27 x 6 [1M]
## # Key:       .model [1]
##    .model          date             value .mean                  `80%`
##    <chr>          <mth>            <dist> <dbl>                 <hilo>
##  1 arima012111 2020 Jan   t(N(5, 0.0023))  149. [140.4482, 158.7148]80
##  2 arima012111 2020 Feb t(N(4.8, 0.0023))  122. [114.3904, 129.2783]80
##  3 arima012111 2020 Mar   t(N(5, 0.0024))  147. [138.3479, 156.7345]80
##  4 arima012111 2020 Apr t(N(5.1, 0.0025))  157. [146.8242, 166.7344]80
##  5 arima012111 2020 May t(N(5.1, 0.0026))  162. [151.8613, 172.8587]80
##  6 arima012111 2020 Jun t(N(5.1, 0.0026))  165. [154.4920, 176.2579]80
##  7 arima012111 2020 Jul t(N(5.1, 0.0027))  159. [148.8580, 170.2144]80
##  8 arima012111 2020 Aug   t(N(5, 0.0028))  147. [137.3038, 157.3518]80
##  9 arima012111 2020 Sep   t(N(5, 0.0029))  149. [138.4153, 158.9728]80
## 10 arima012111 2020 Oct  t(N(5.1, 0.003))  165. [153.5065, 176.6852]80
## # ... with 17 more rows, and 1 more variable: `95%` <hilo>
```

```r
print(arima_point_intervals)
```

```
## # A tsibble: 27 x 6 [1M]
## # Key:       .model [1]
##    .model          date             value .mean                  `80%`
##    <chr>          <mth>            <dist> <dbl>                 <hilo>
##  1 arima012111 2020 Jan   t(N(5, 0.0023))  149. [140.4482, 158.7148]80
##  2 arima012111 2020 Feb t(N(4.8, 0.0023))  122. [114.3904, 129.2783]80
##  3 arima012111 2020 Mar   t(N(5, 0.0024))  147. [138.3479, 156.7345]80
##  4 arima012111 2020 Apr t(N(5.1, 0.0025))  157. [146.8242, 166.7344]80
##  5 arima012111 2020 May t(N(5.1, 0.0026))  162. [151.8613, 172.8587]80
##  6 arima012111 2020 Jun t(N(5.1, 0.0026))  165. [154.4920, 176.2579]80
##  7 arima012111 2020 Jul t(N(5.1, 0.0027))  159. [148.8580, 170.2144]80
##  8 arima012111 2020 Aug   t(N(5, 0.0028))  147. [137.3038, 157.3518]80
##  9 arima012111 2020 Sep   t(N(5, 0.0029))  149. [138.4153, 158.9728]80
## 10 arima012111 2020 Oct  t(N(5.1, 0.003))  165. [153.5065, 176.6852]80
## # ... with 17 more rows, and 1 more variable: `95%` <hilo>
```

The training set that consists of the dates 1982 to 2019 will be used to forecast the test set that contains the period post 2019. Since the sample ends in March 2022, the forecasted value used will be 27 months.

The best benchmark method from previous assignment was the seasonal naïve model. Forecasting this model onto to the observed data shows that the forecast intervals capture the seasonality peaks, however, did not capture the cyclic trends. Therefore, the forecasts intervals are a bit wide. The point intervals for SNAIVE model are also quite wide for both the 80% and 95% prediction intervals, indicating a lot of residuals left over.

The ETS(M,A,M) model was also able to capture the seasonality pattern but also was not able to capture the cyclic pattern around the first half of the year. The forecast intervals are also wide in this model, but the seasonality peaks the forecast intervals are narrow as they closely align with the original.

The ARIMA(0,1,2)(1,1,1) model also captured the seasonality pattern but fail to capture the cyclic patterns again. However, the forecast intervals are much narrower than the other two models. The seasonality peak around December 2019 closely aligns with the original data. The 95% prediction intervals for the end of year are somewhat narrower than the previous models as well.

Question 10

Evaluate the accuracy of the point forecasts over the period post 2019. A table with accuracy measures will be necessary to be presented here. Comment on which forecasts are the most accurate. (4 marks)

```
accuracy(abs_data_benchmark_fc,abs_data_test)
```

```
## # A tibble: 1 x 10
##   .model         .type    ME  RMSE   MAE   MPE  MAPE  MASE RMSSE  ACF1
##   <chr>          <chr> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 Seasonal_naive Test   18.0  23.4  21.0  9.12  11.4   NaN   NaN 0.256
```

```
accuracy(ETS_fc, abs_data_test)
```

```
## # A tibble: 1 x 10
##   .model   .type    ME  RMSE   MAE   MPE  MAPE  MASE RMSSE  ACF1
##   <chr>    <chr> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 hw_multi Test   18.0  23.8  21.5  9.04  11.7   NaN   NaN 0.357
```

```
accuracy(arima_fc, abs_data_test)
```

```
## # A tibble: 1 x 10
##   .model      .type    ME  RMSE   MAE   MPE  MAPE  MASE RMSSE  ACF1
##   <chr>       <chr> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 arima012111 Test   15.8  21.7  19.4  7.84  10.6   NaN   NaN 0.319
```

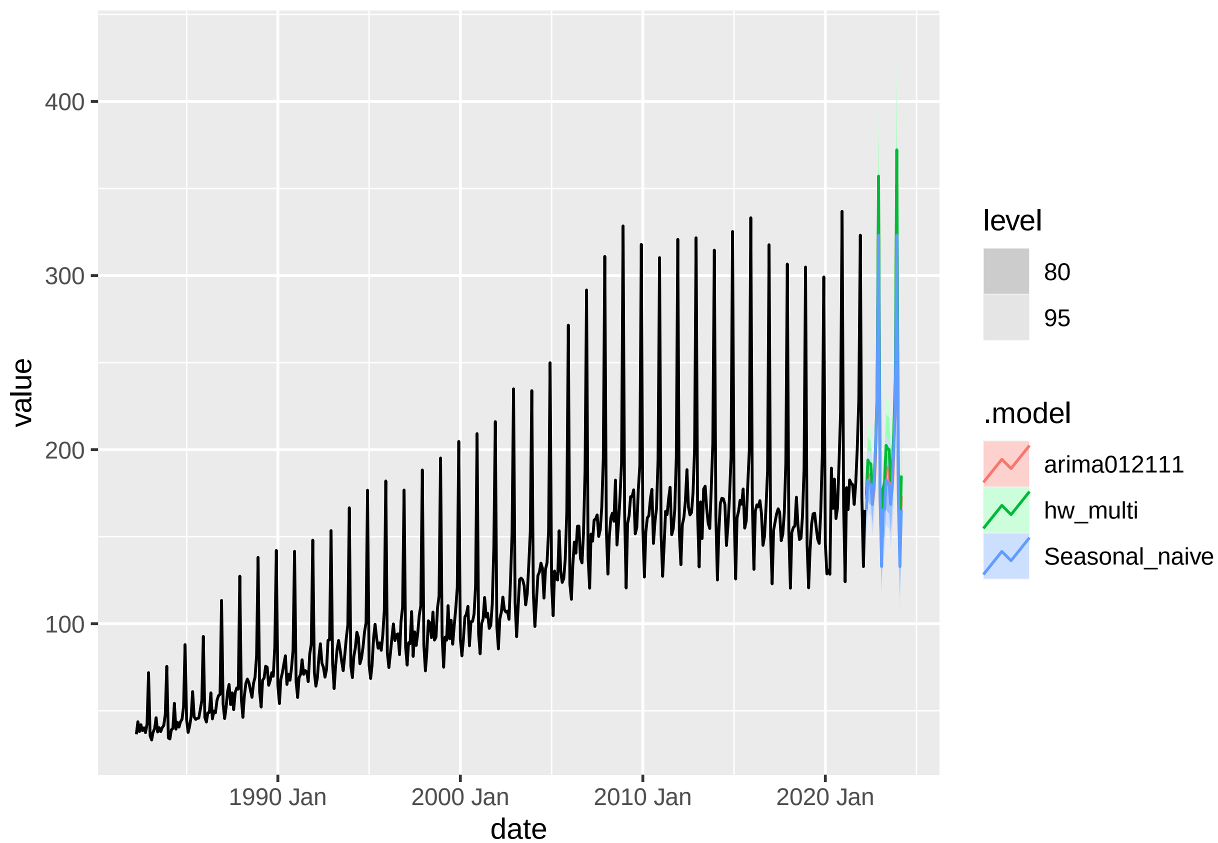
The RMSE value for the ARIMA model outperforms both the seasonal naive and ETS(M,A,M) model. Hence the ARIMA model would be a better fitted model as it has the smallest magnitude of errors.

Question 11

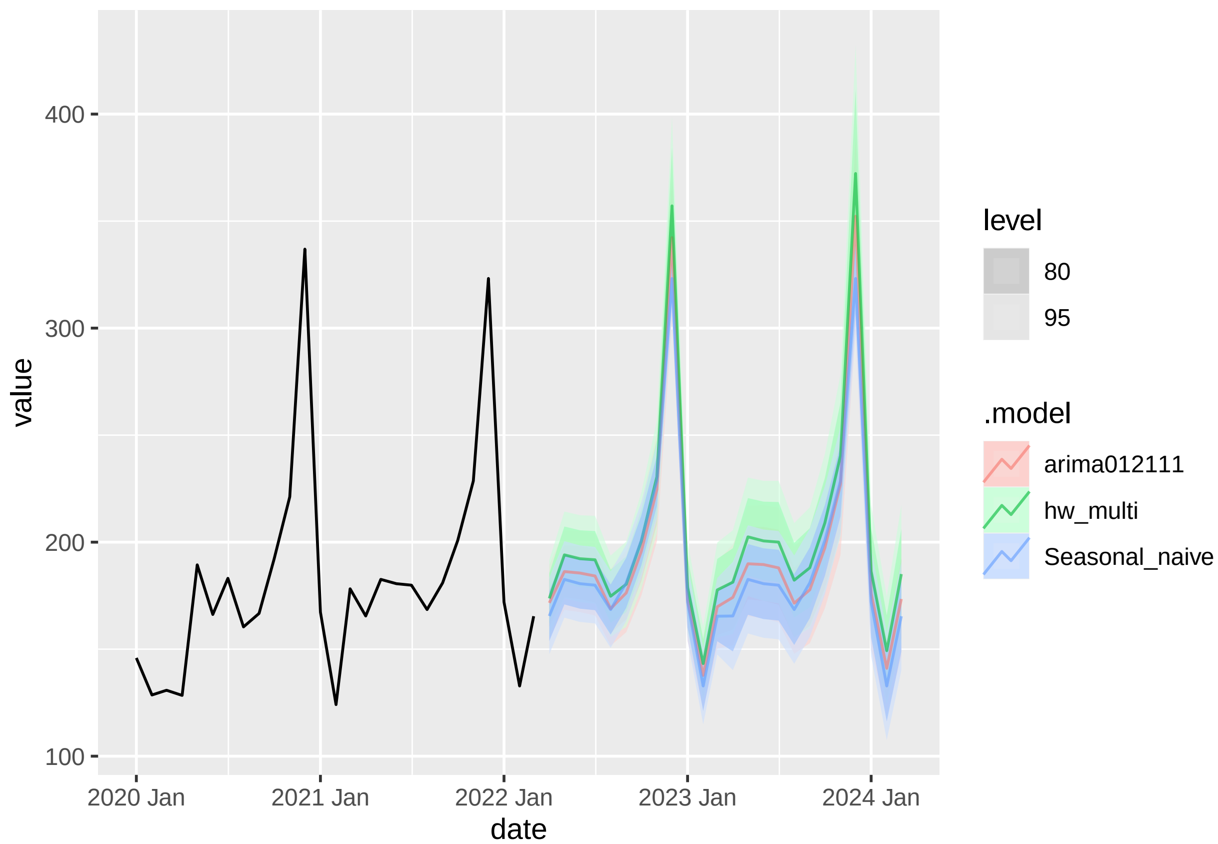
Use all three models to forecast the next 24-months of your updated series. Generate the necessary plots and comment on the forecasts. Make sure you can clearly visualise these. How have your models faired amidst the effects of COVID-19? (9 marks)

```
all_model_fit <- abs_data %>%
  model(
    Seasonal_naive = SNAIVE(value),
    arima012111 = ARIMA(log(value) ~ pdq(0,1,2) + PDQ(1,1,1)),
    hw_multi = ETS(value ~ error("M") + trend("A") +
                                                    season("M"))
        )
all_model_fc <- all_model_fit %>%
  forecast(h = '24 months')

all_model_fc %>%
  autoplot(abs_data)
```




```
all_model_fc  %>%
  autoplot(filter(abs_data, year(date)>=2020), alpha=0.6)
```



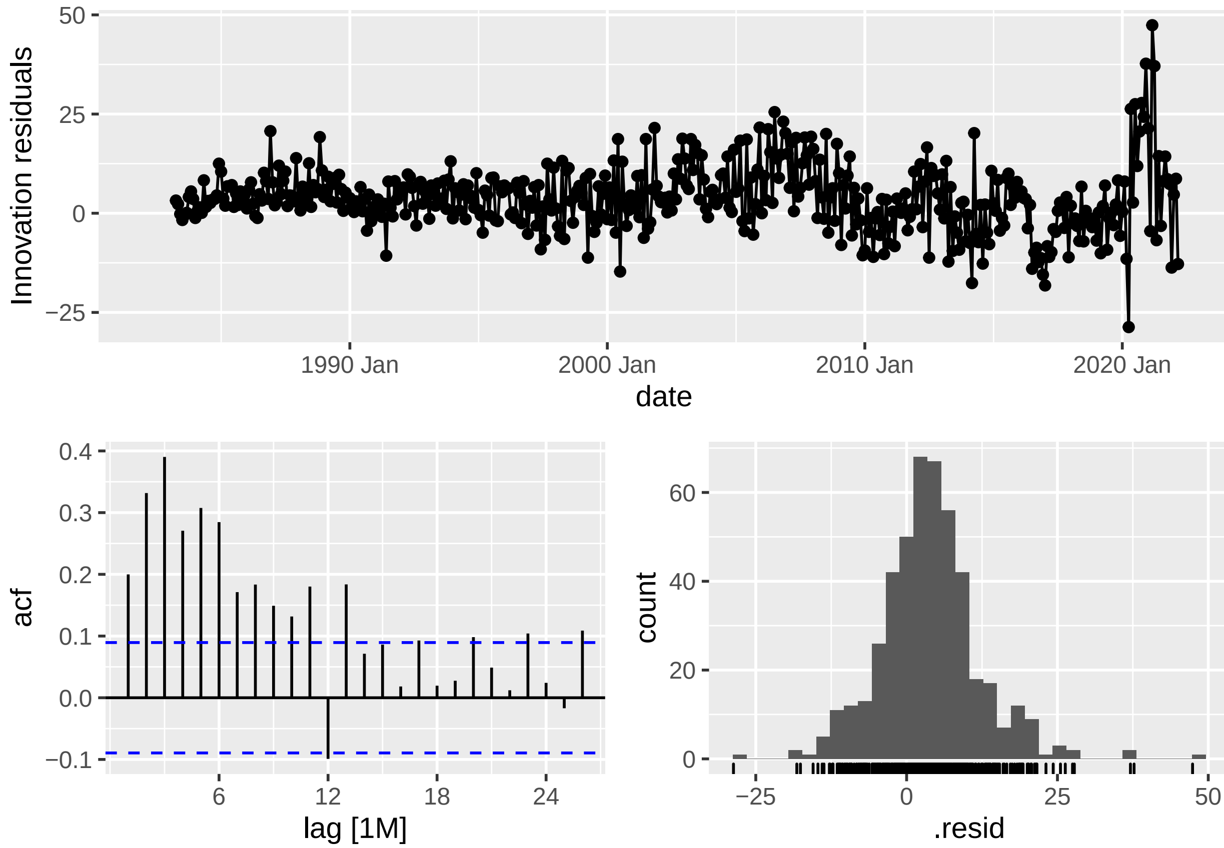

```
point_intervals <- all_model_fit %>%
  forecast(h = '24 months') %>%
  hilo()
view(point_intervals)


all_model_fit %>%
  select(Seasonal_naive) %>%
  gg_tsresiduals()
```

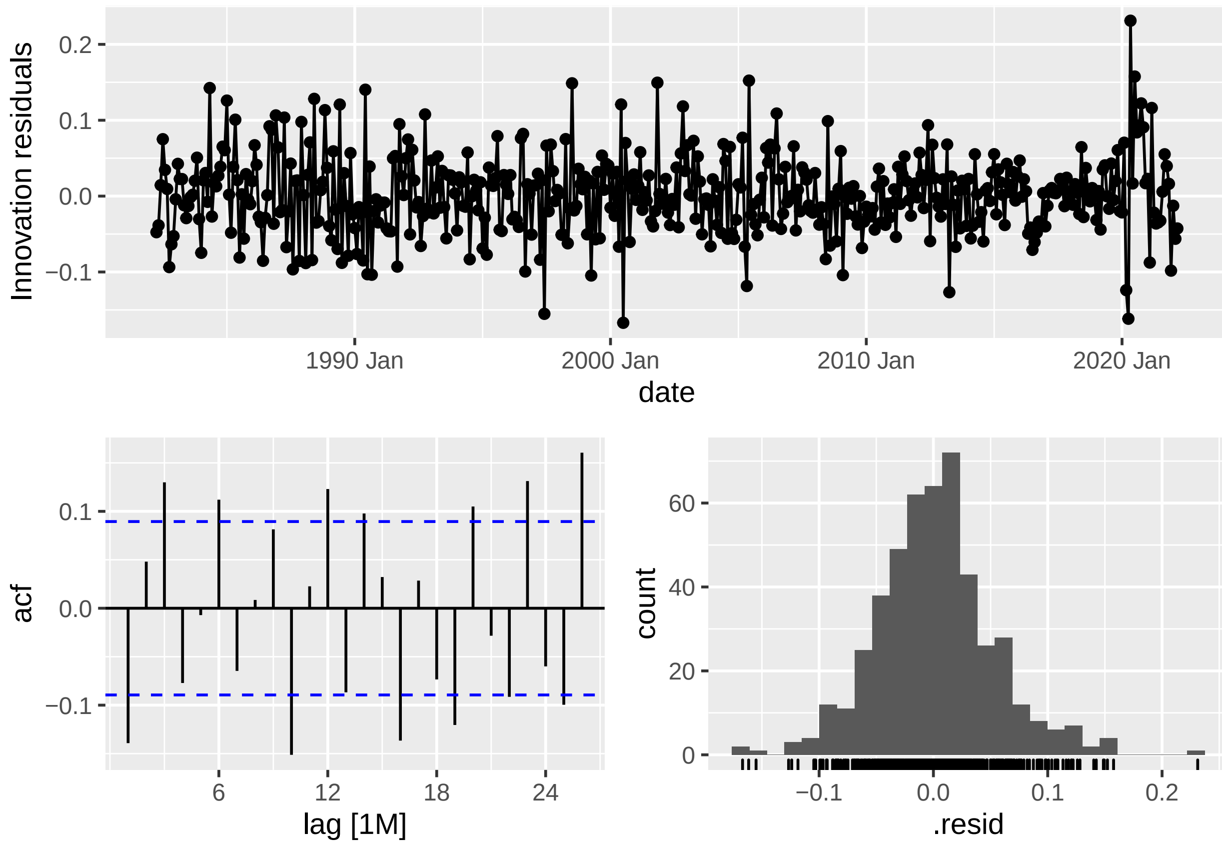
```
## Warning: Removed 12 row(s) containing missing values (geom_path).

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

## Warning: Removed 12 rows containing non-finite values (stat_bin).
```




```
all_model_fit %>%
  select(hw_multi) %>%
  gg_tsresiduals()
```



The ETS(M,A,M) forecast intervals are the widest compare to the other two, whereas the seasonal naive model has the narrowest forecast interval. The prediction intervals for the seasonality naive method are also narrower.

Undergoing a residual check shows that there are less white noise and residuals for the seasonal naive method compared to the ETS(M,A,M) model. But the forecast and prediction intervals are still wide on some cases due to the pandemic being unpredictable. For instance, the seasonality peaks in December 2023 are quite narrow and closely align together. But a further forecast in December 2024 show that the intervals are wider.

In saying that all 3 models seem to capture the seasonality pattern quite well. The models are also able to capture the small upward and downward trends in the early in the year. These models are capturing that even amidst the COVID-19 pandemic, the turnover of Western Australia department stores will continue to peak at the end of each year. The ARIMA and ETS(M,A,M) model especially, show that the turnover is expected to increase as the peaks are higher than 2021 and 2022.

You have now completed all tasks for this unit if you are enrolled in ETF3231 (one more to go for ETF5231). I hope you have learnt a lot from your hard work throughout the semester completing the assignments.

Many congratulations.
Cheers,
George