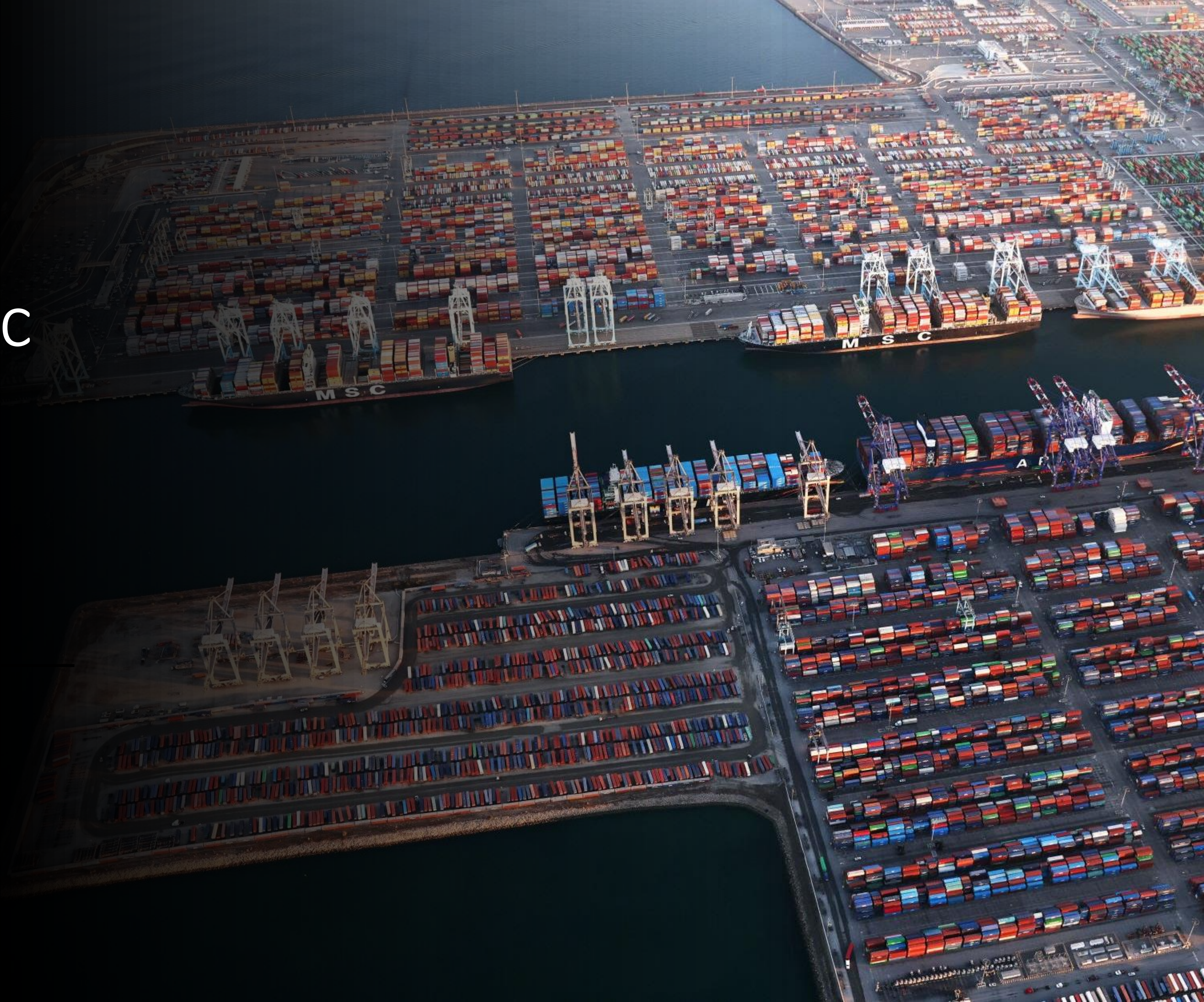
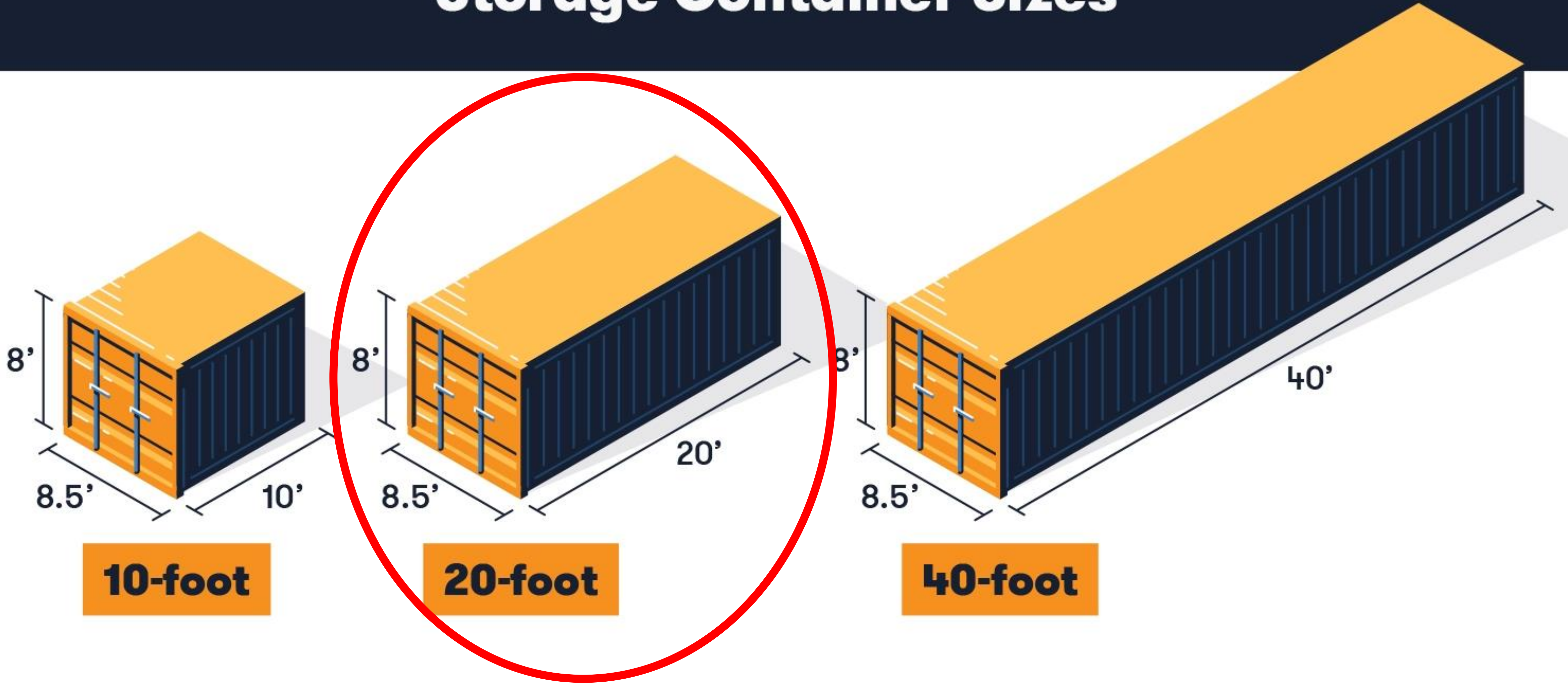
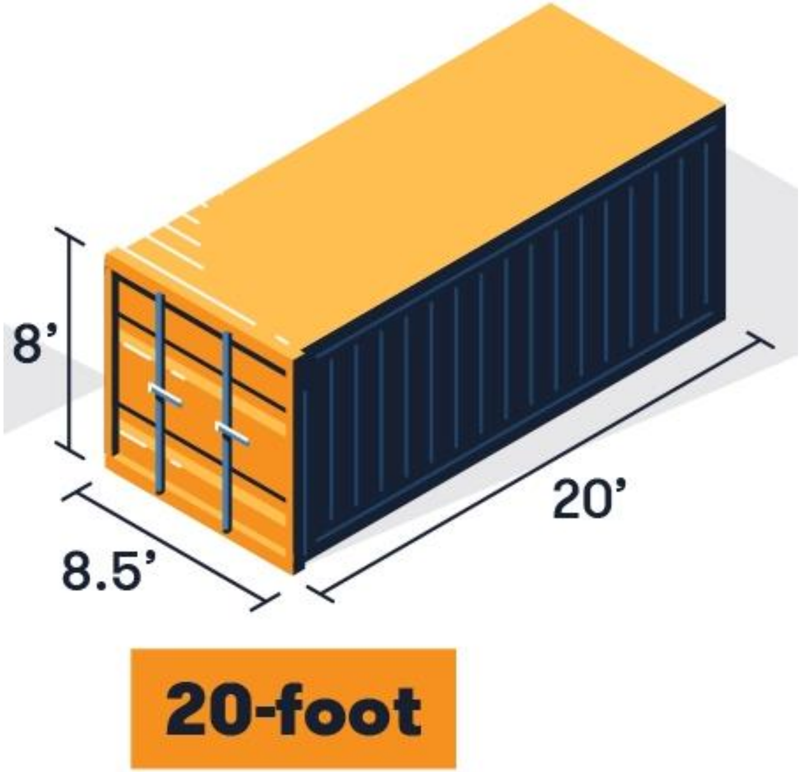
Forecasting Container Traffic for the Port of Los Angeles



By: Jason Rogers and Daniel Schumacher

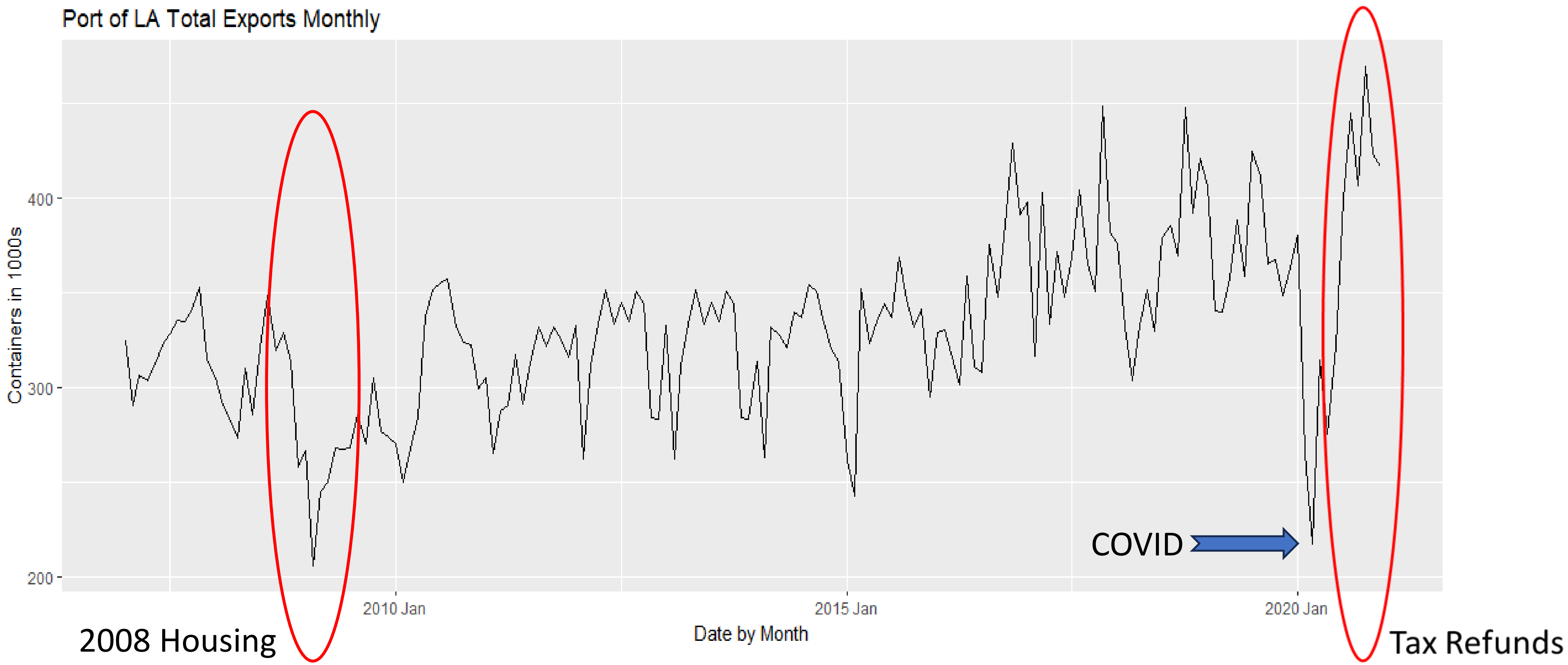
Storage Container Sizes



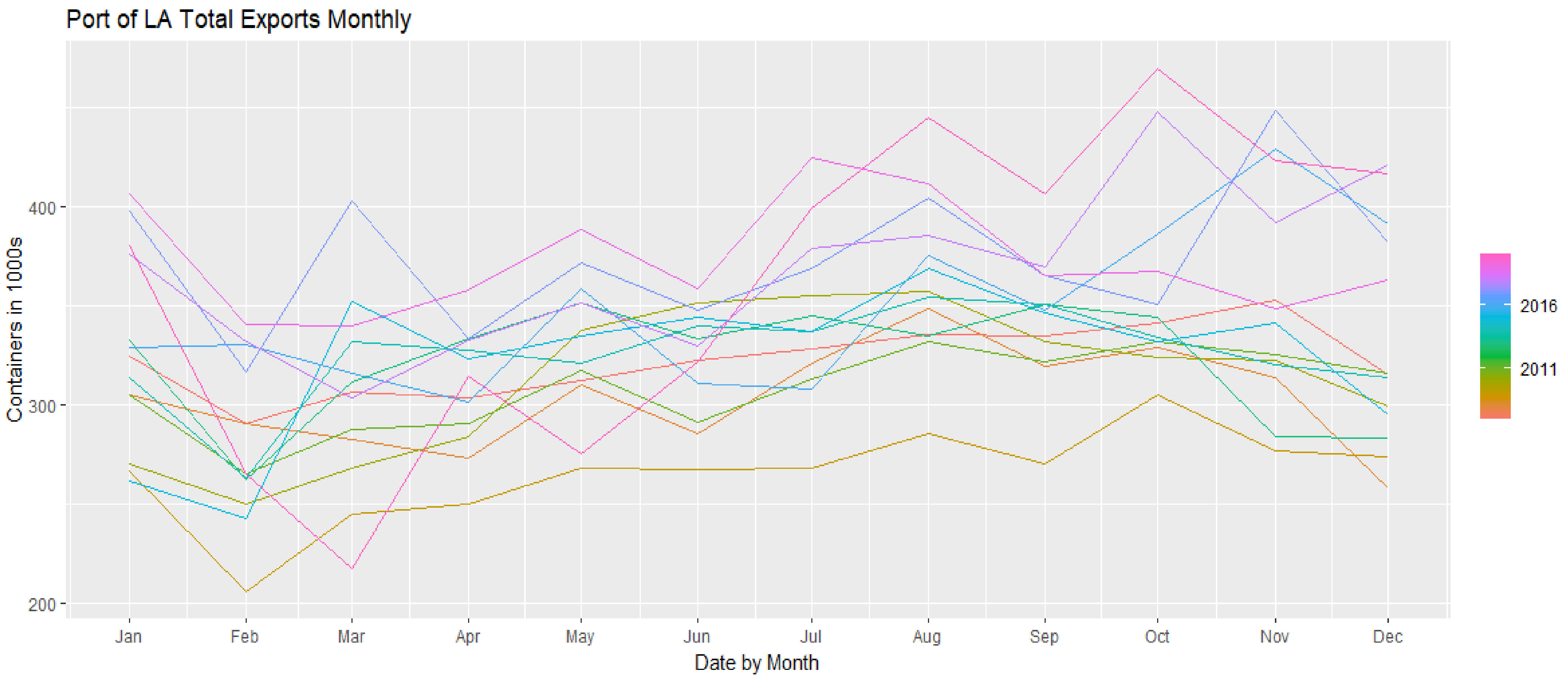


= Twenty Foot Equivalent Units (TEUs)

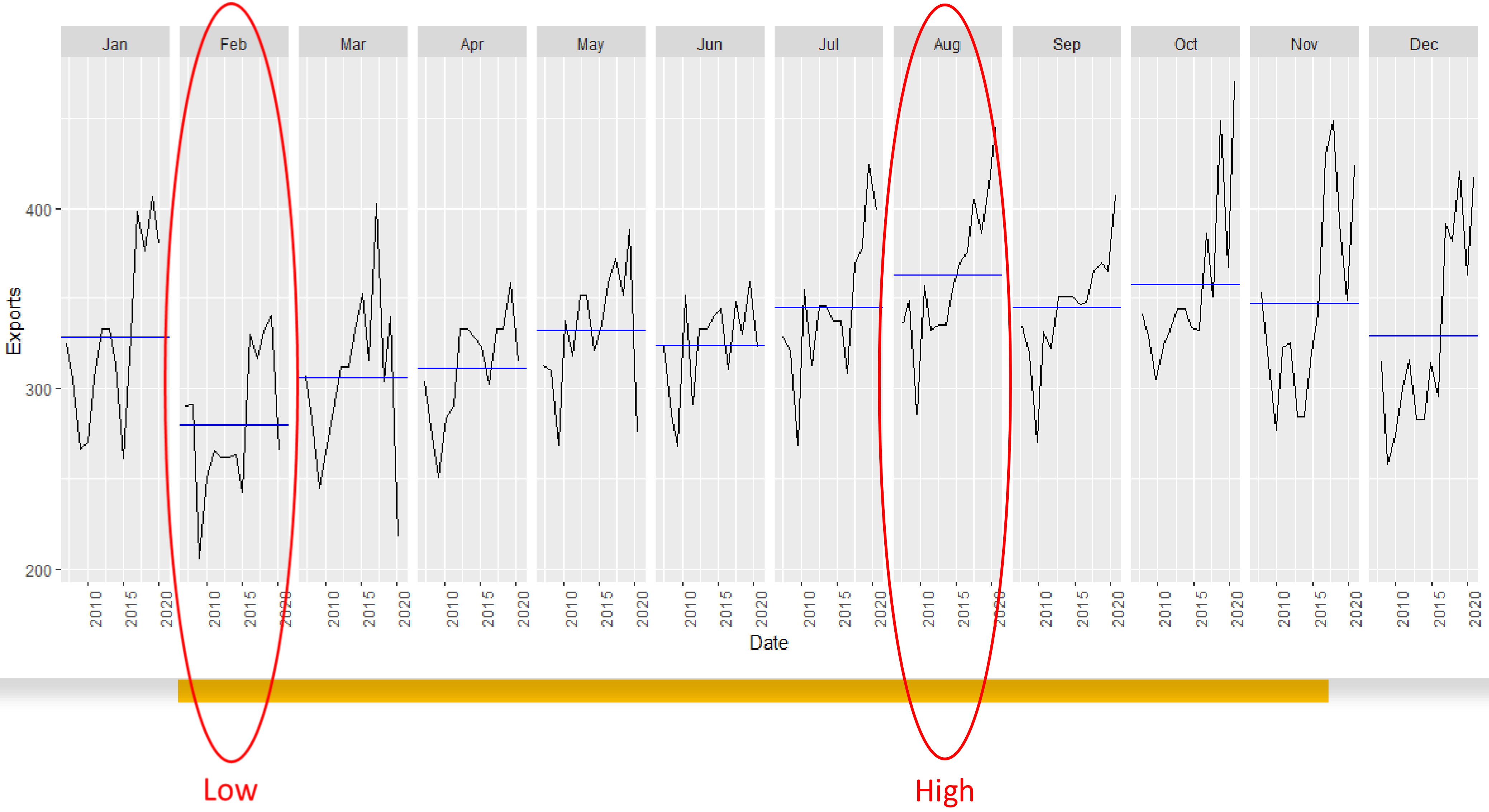
Data Overview – Total Exports



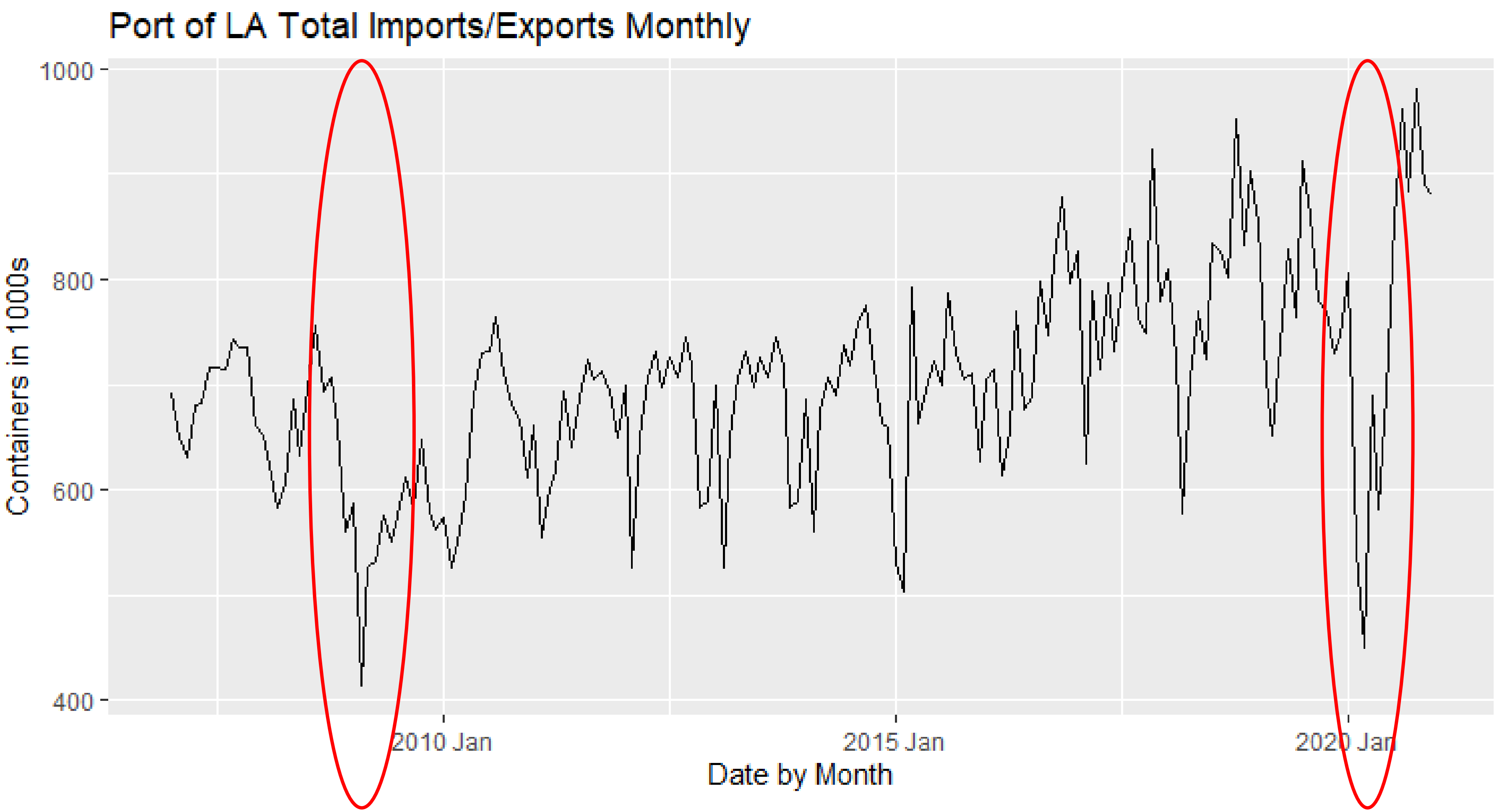
Seasonality Overview – Total Exports



Seasonality Overview – Total Exports



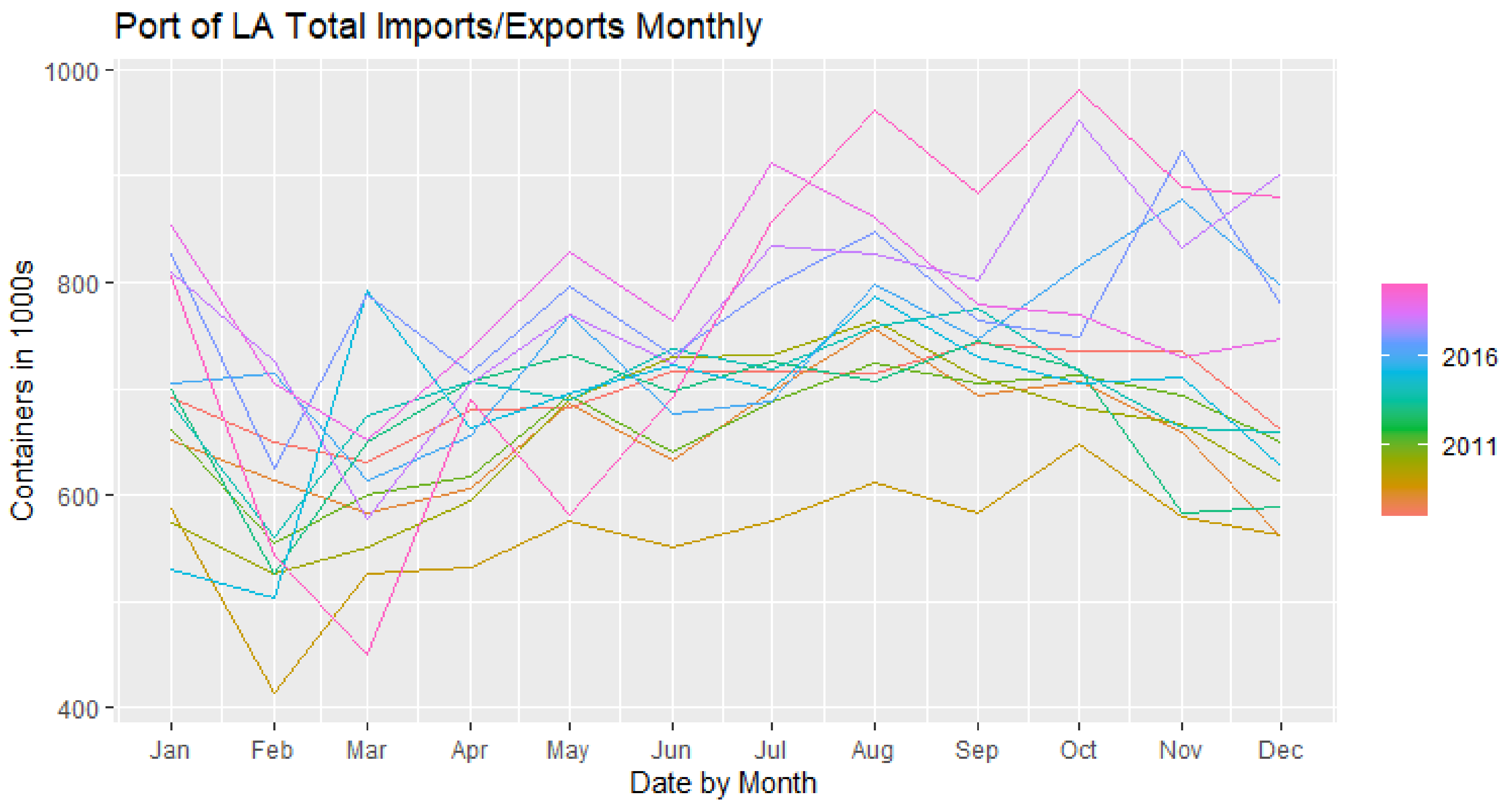
Data Overview – Total TEUs



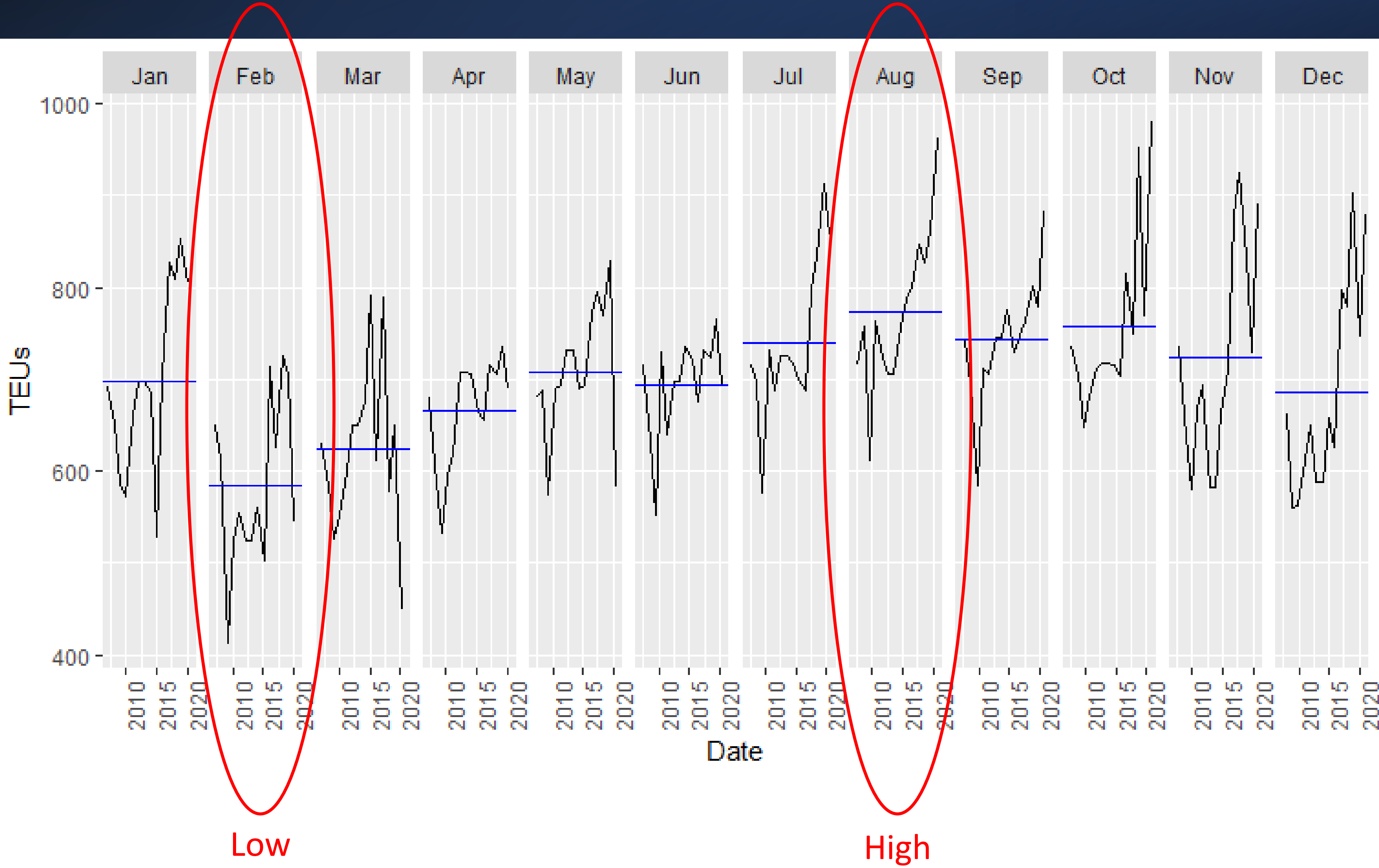
2008 Housing

COVID-19

Seasonality Overview – Total TEUs



Seasonality Overview – Total TEUs



Compare Linear Models

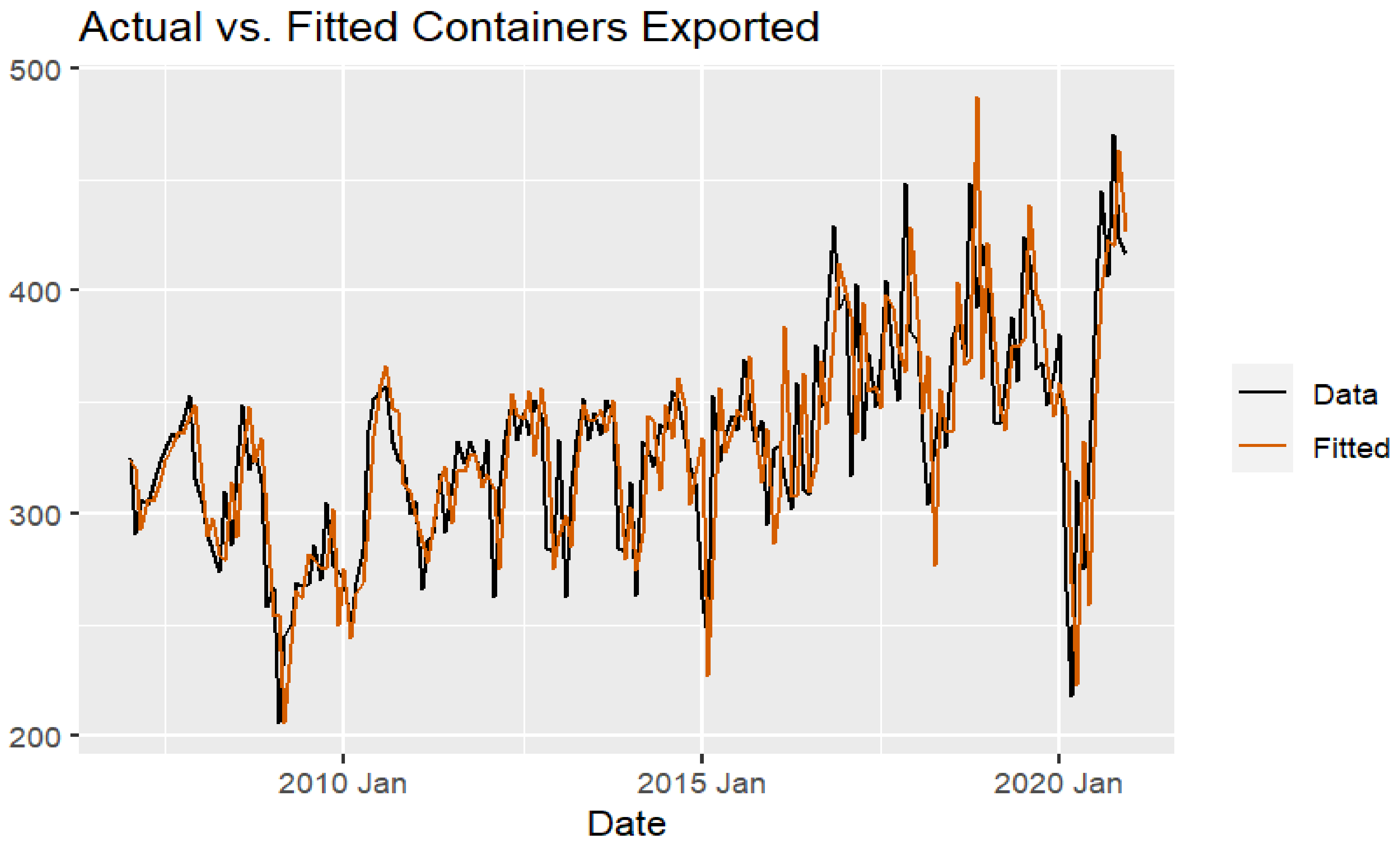
Total Exports:

Linear Trend selected with residual standard error 32.85

Total TEUs:

Quadratic Trend selected with residual standard error 69.11 v 71.09

ARIMA Fit – Total Exports



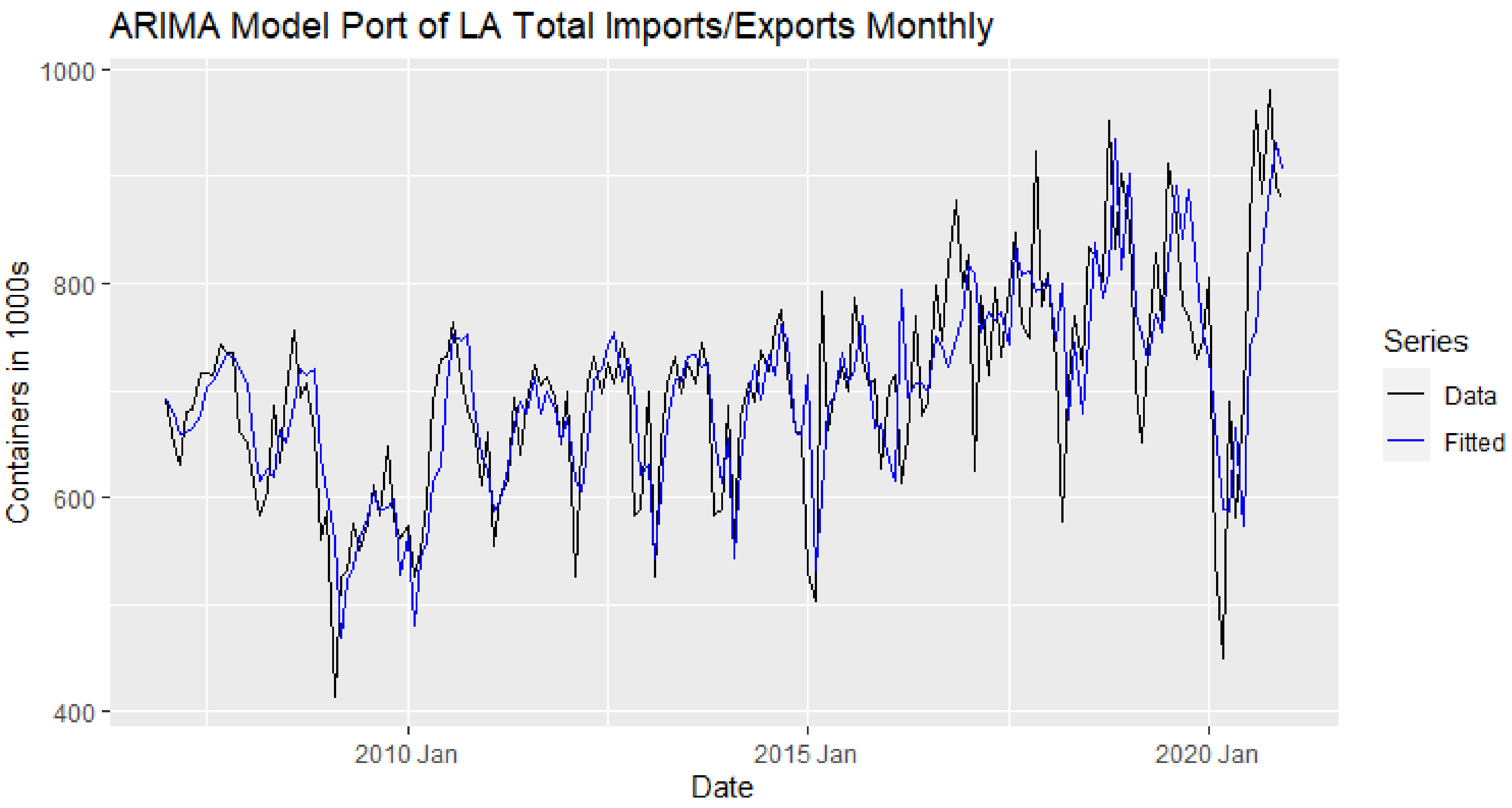
Selected Model
pdq (0,1,0) + PDQ (1,0,1)
with an AIC = 1644.33

ARIMA Fit – Total TEUs

Selected Model

pdq (2,1,0) + PDQ (1,0,1)

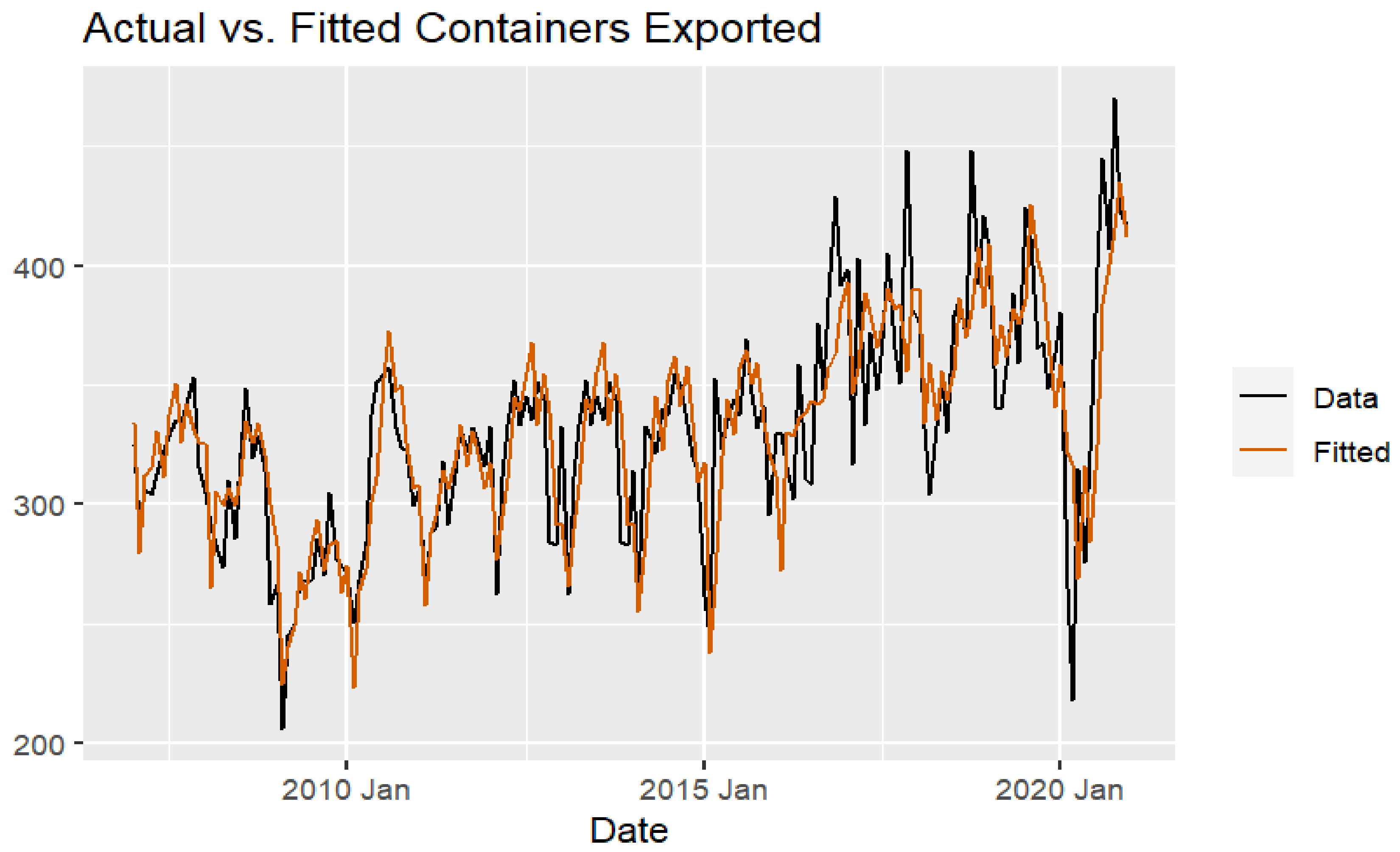
with an AIC = 1883.36



ETS Fit – Total Exports

ETS model fits best with A, A, A or data that has constant trend with additive seasonality.

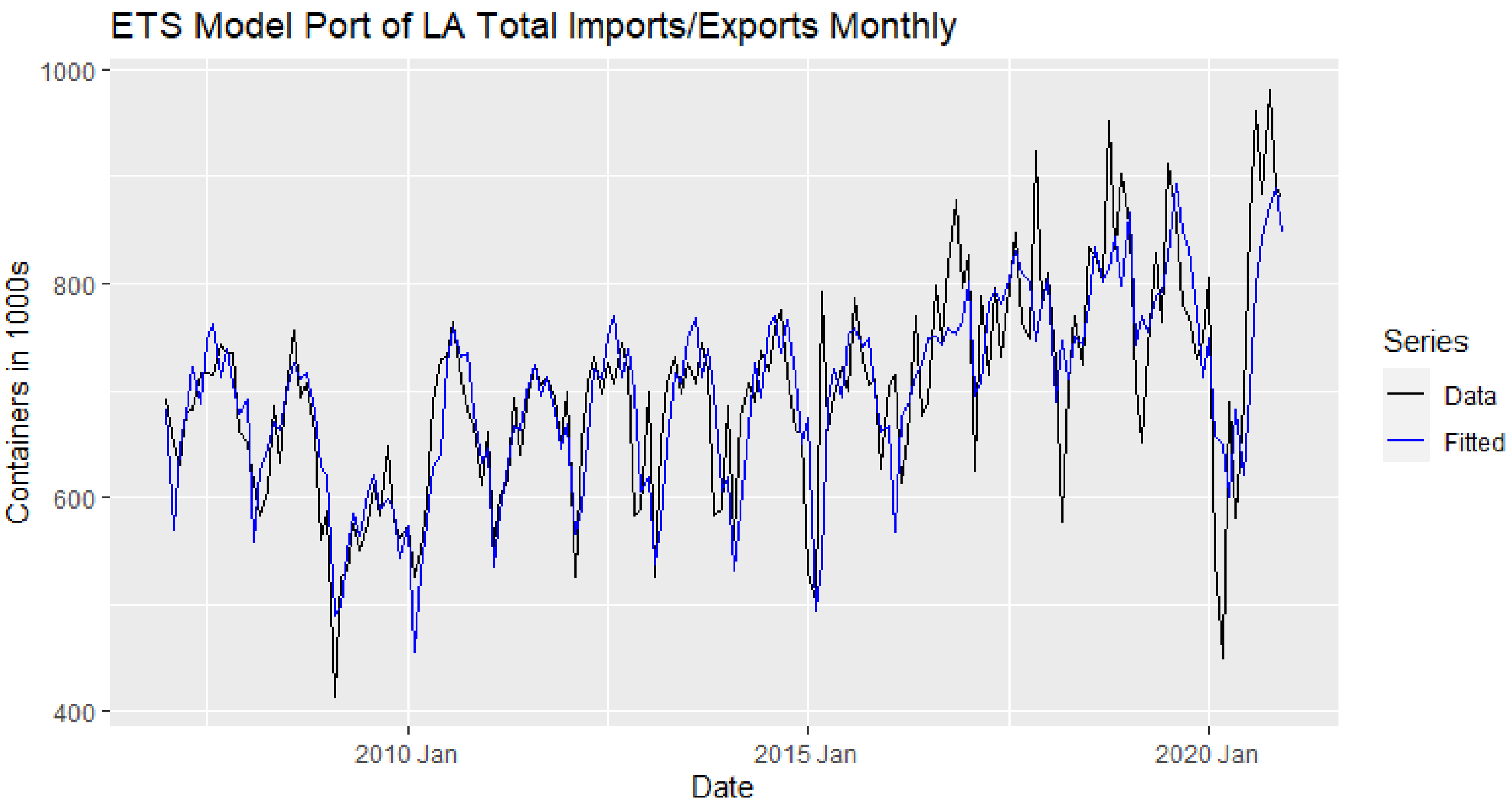
AIC = 2002.62



ETS Fit – Total TEUs

ETS model fits with best with A, N, A or data that has constant trend with additive seasonality.

AIC = 2268.76



Comparing All Models

Total Exports:

Best performing model was auto ETS

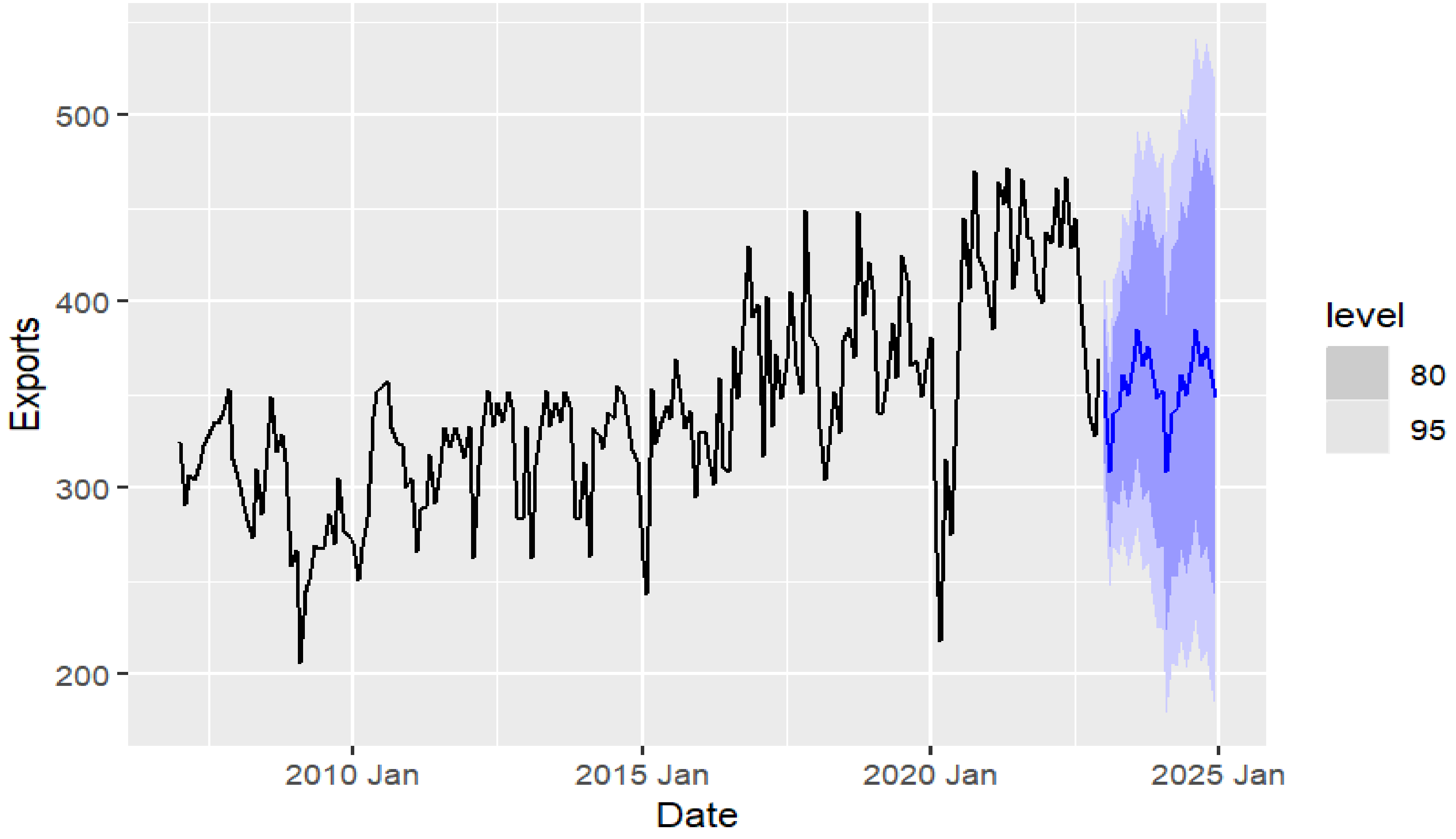
	.model	ME	RMSE	MAE	MPE	MAPE	MASE	RMSSE
1	ets_auto	-0.437	47.2	37.6	-1.16	9.40	1.22	1.16
2	ets_manual1	-7.27	50.2	38.3	-2.86	9.75	1.25	1.23
3	combination	3.89	53.0	41.7	-0.249	10.4	1.36	1.30
4	ARIMA.auto	-10.3	60.9	44.3	-3.87	11.3	1.44	1.49
5	ARIMA3.fit	e0.223	59.4	46.3	-1.25	11.5	1.50	1.45
6	train.tb_lm	37.3	62.2	54.7	7.89	13.0	1.78	1.52

Total TEUs:

Best performing model was ETS A, N, A

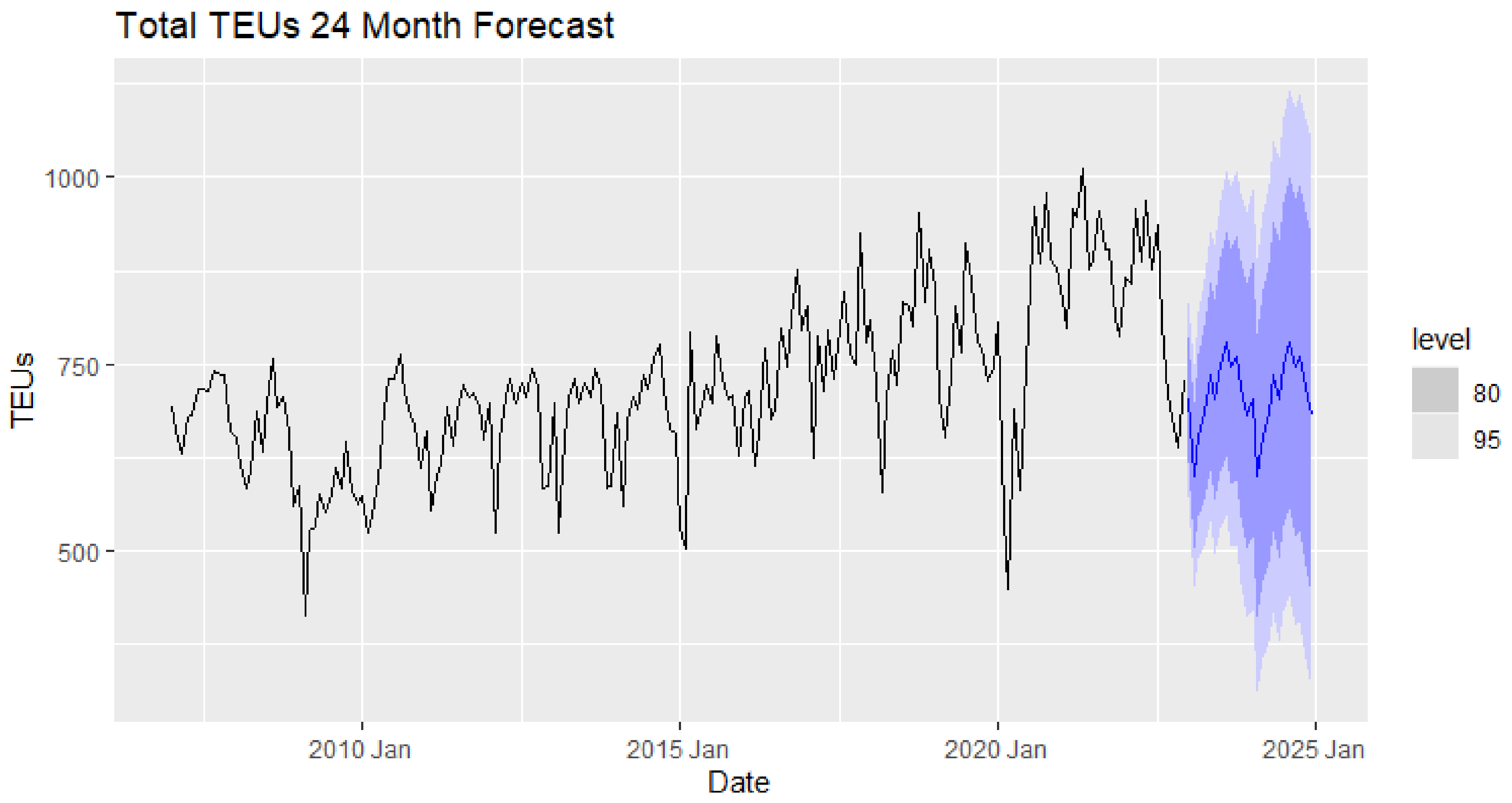
	.model	ME	RMSE	MAE	MPE	MAPE	MASE	RMSSE
1	ETS_manual	-22.7	116	87.6	-4.17	11.1	1.35	1.34
2	ets_auto	-22.7	116	87.6	-4.17	11.1	1.35	1.34
3	ts_reg	3.49	125	95.6	-1.21	11.9	1.47	1.44
4	combination	-22.9	132	99.5	-4.39	12.5	1.53	1.53
5	arima_auto	-20.1	152	114.	-4.32	14.2	1.76	1.76
6	arima_manual	52.3	160	125.	-8.11	15.8	1.93	1.85

Total Exports 24 Month Forecast



2023 Jan	252.27
2023 Feb	308.24
2023 Mar	340.00
2023 Apr	342.54
2023 May	360.65
2023 Jun	349.30
2023 Jul	367.55
2023 Aug	385.31
2023 Sep	365.83
2023 Oct	375.47
2023 Nov	363.90
2023 Dec	348.09
2024 Jan	352.27
2024 Feb	308.24
2024 Mar	340.00
2024 Apr	342.54
2024 May	360.65
2024 Jun	349.30
2024 Jul	367.55
2024 Aug	385.31
2024 Sep	365.83
2024 Oct	375.47
2024 Nov	363.90
2024 Dec	348.09

Total TEUs 24 Month Forecast

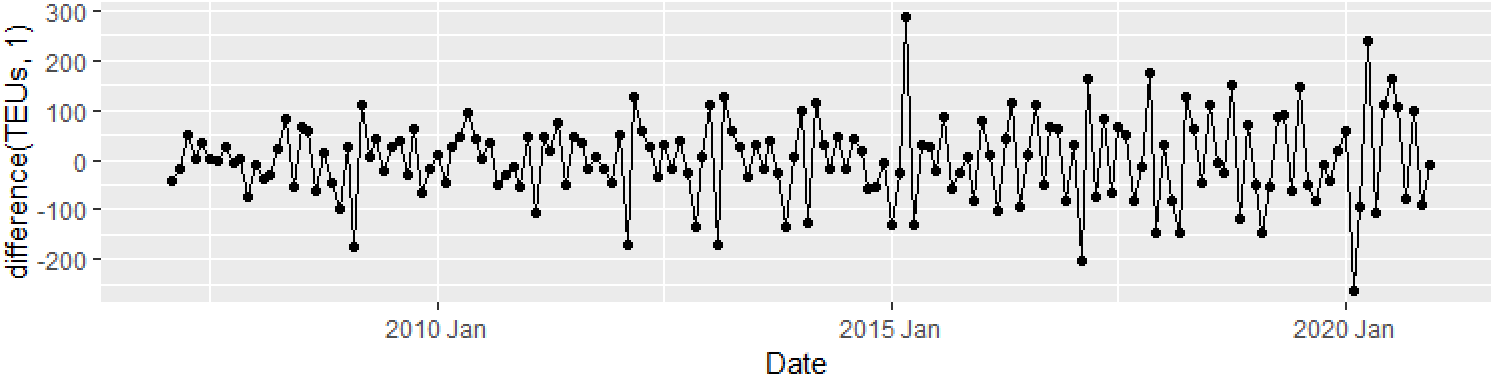


2023 Jan	703.1859
2023 Feb	600.1623
2023 Mar	654.8572
2023 Apr	684.1826
2023 May	734.5271
2023 Jun	702.0776
2023 Jul	749.1346
2023 Aug	778.7497
2023 Sep	746.2127
2023 Oct	758.8937
2023 Nov	718.5186
2023 Dec	681.6405
2024 Jan	703.1859
2024 Feb	600.1623
2024 Mar	654.8572
2024 Apr	684.1826
2024 May	734.5271
2024 Jun	702.0776
2024 Jul	749.1346
2024 Aug	778.7497
2024 Sep	746.2127
2024 Oct	758.8937
2024 Nov	718.5186
2024 Dec	681.6405

Questions

Appendix A: ARIMA Fit – Total Exports

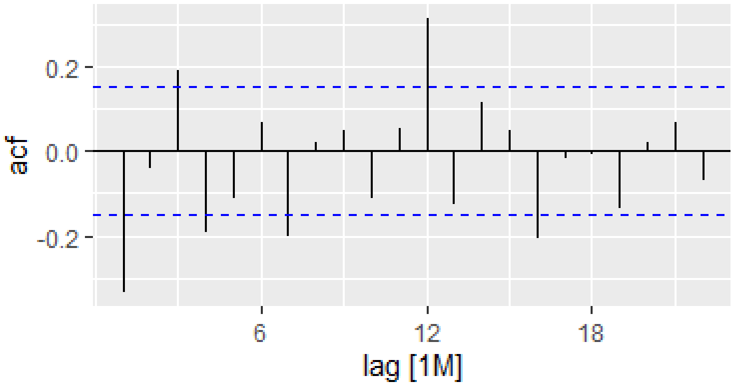
Plotting ACF and PACF

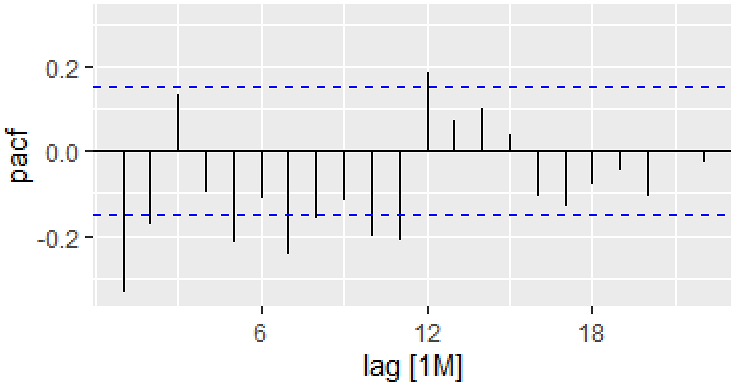


Testing for P = 1 due to significant value at PACF = 1
pdq (0,1,0) + PDQ (1,0,0)
AIC = 1932.34

Testing for Q = 1 due to significant value at ACF = 1
pdq (0,1,0) + PDQ (0,0,1)
AIC = 1942.94

Testing for Both P = 1 and Q =1
pdq (0,1,0) + PDQ (1,0,1)
AIC = 1925.03





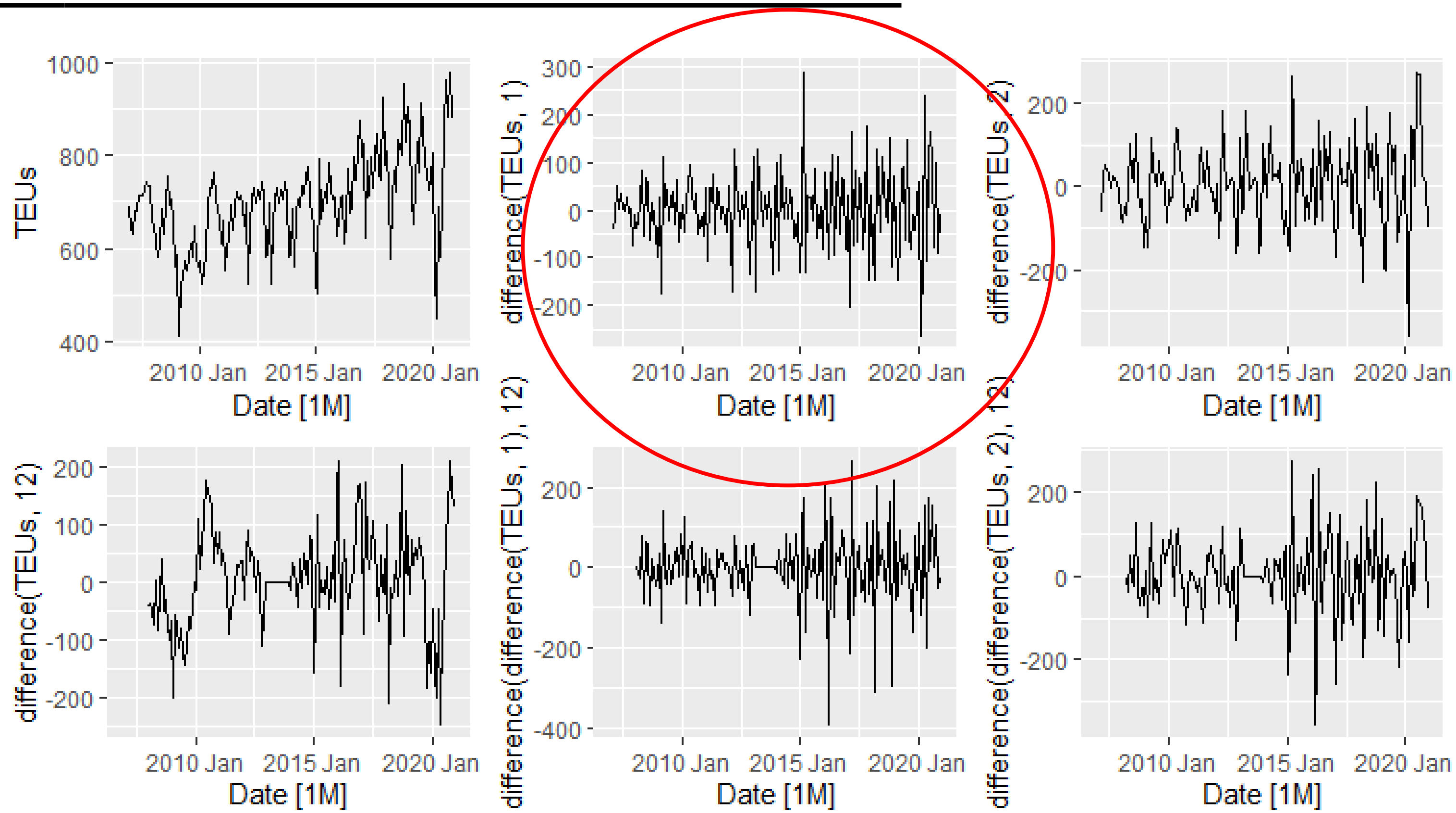
UPDATE

Appendix B: ARIMA Fit – Total TEUs

Test for d and D:

Do we need to remove any seasonal trend? (D)
TEUs.train %>% features(TEUs, unitroot_nsdiffs)
Do not test for seasonal (lag 12) difference due to the results being 0, D=0

Do we also need to remove any non-seasonal trend? (d) TEUs.train %>% features(TEUs, unitroot_ndiffs)
The number 12 due to monthly data for 1 LAG, d = 1.



Graphing the differences confirms d/D test, the circled graph appears to be the smoothest plot

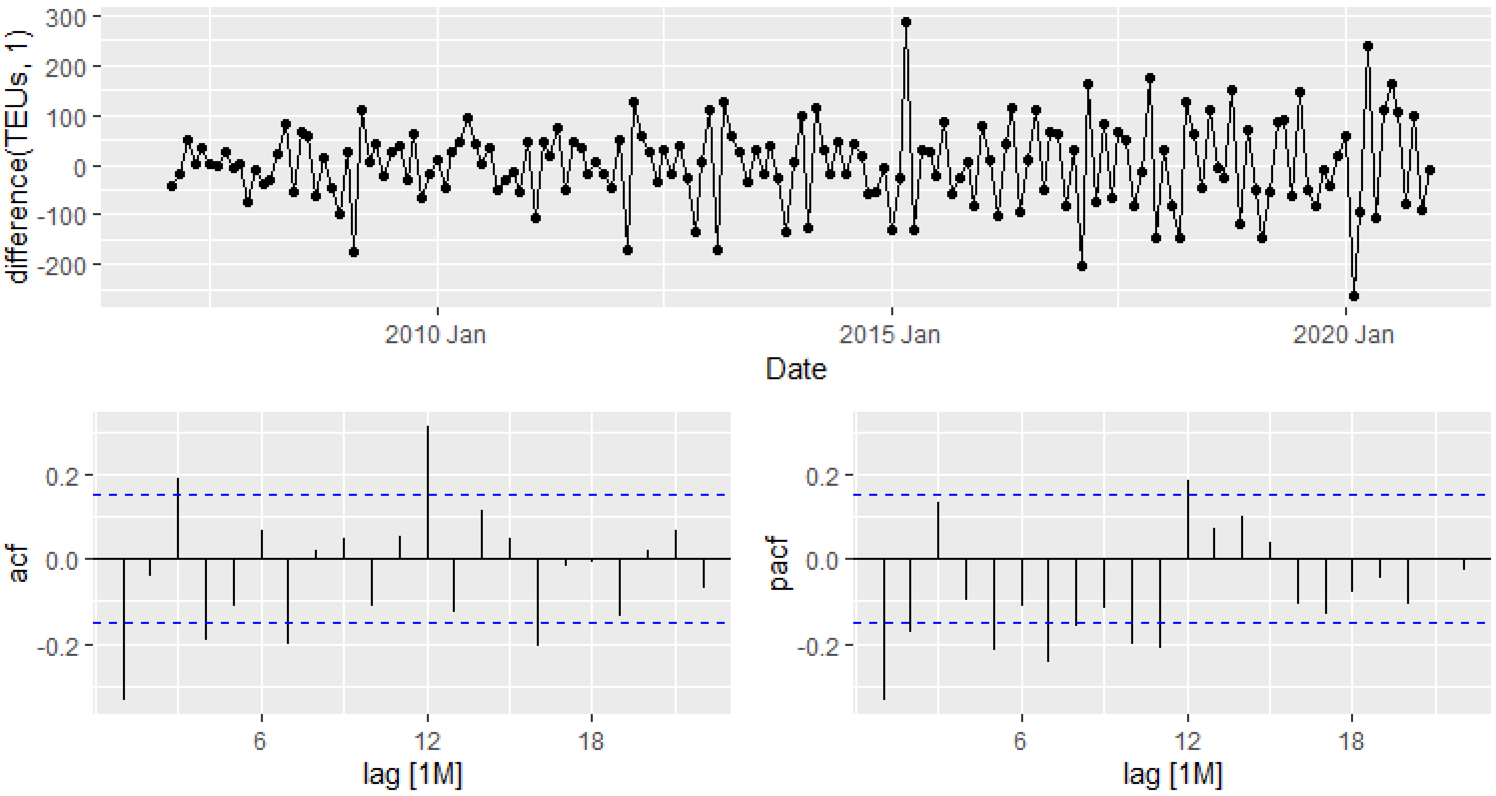
Appendix B: ARIMA Fit – Total TEUs

Plotting ACF and PACF

Testing for P = 1 due to significant value at PACF = 1
pdq (0,1,0) + PDQ (1,0,0)
AIC = 1932.34

Testing for Q = 1 due to significant value at ACF = 1
pdq (0,1,0) + PDQ (0,0,1)
AIC = 1942.94

Testing for Both P = 1 and Q =1
pdq (0,1,0) + PDQ (1,0,1)
AIC = 1925.03



Appendix B: ARIMA Fit – Total TEUs

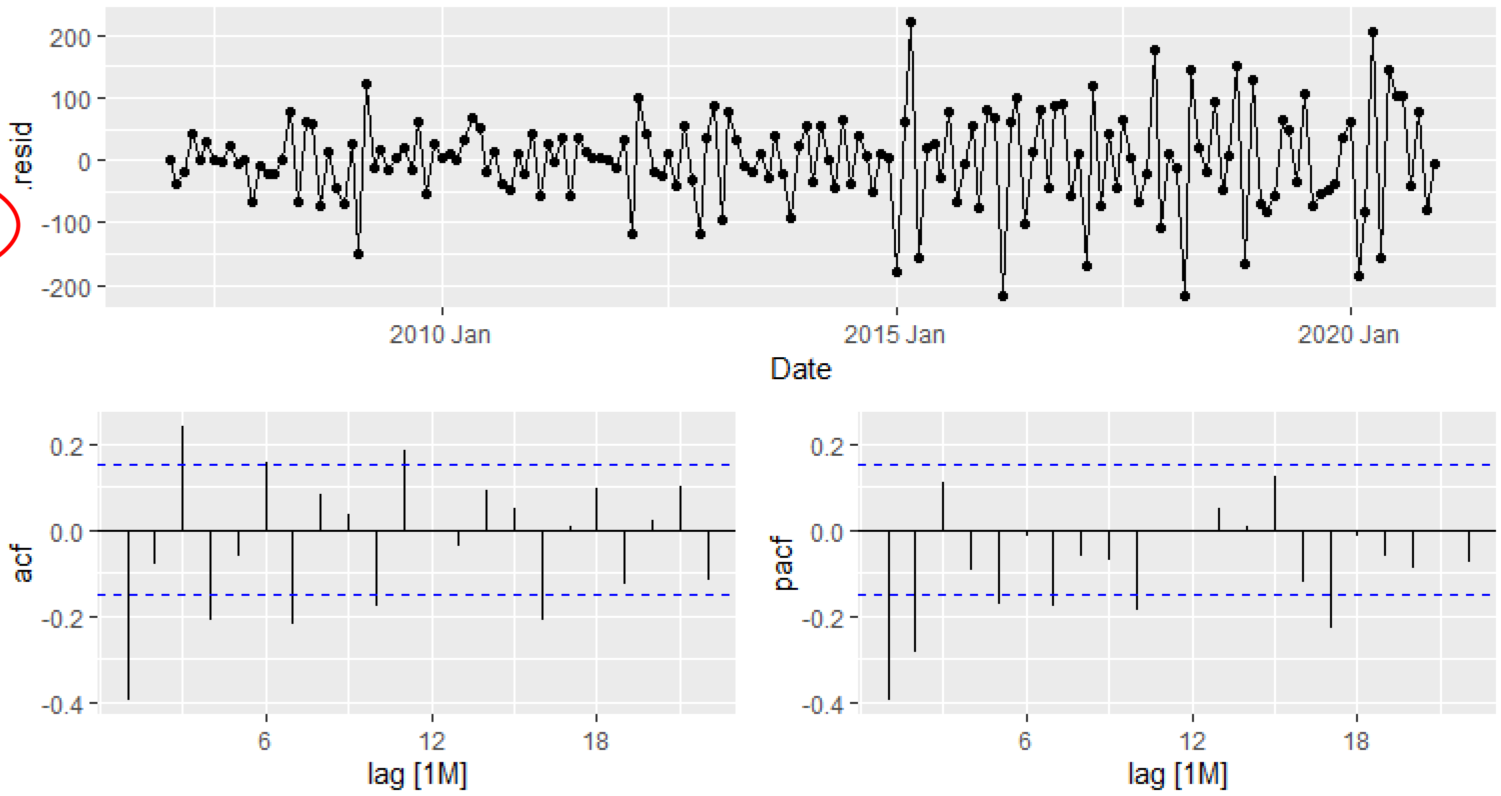
Checking for remaining residuals

Residuals continue to appear at ACF 1 and PACF 2

Testing for p = 2 due to significant value at PACF = 2
pdq (2,1,0) + PDQ (1,0,1)
AIC = 1883.36

Testing for q = 1 due to significant value at ACF = 1
pdq (0,1,1) + PDQ (1,0,1)
AIC = 1885.67

Testing for Both p = 2 and q =1
pdq (2,1,1) + PDQ (1,0,1)
AIC = 1883.56



No Remaining Residuals after running p=2

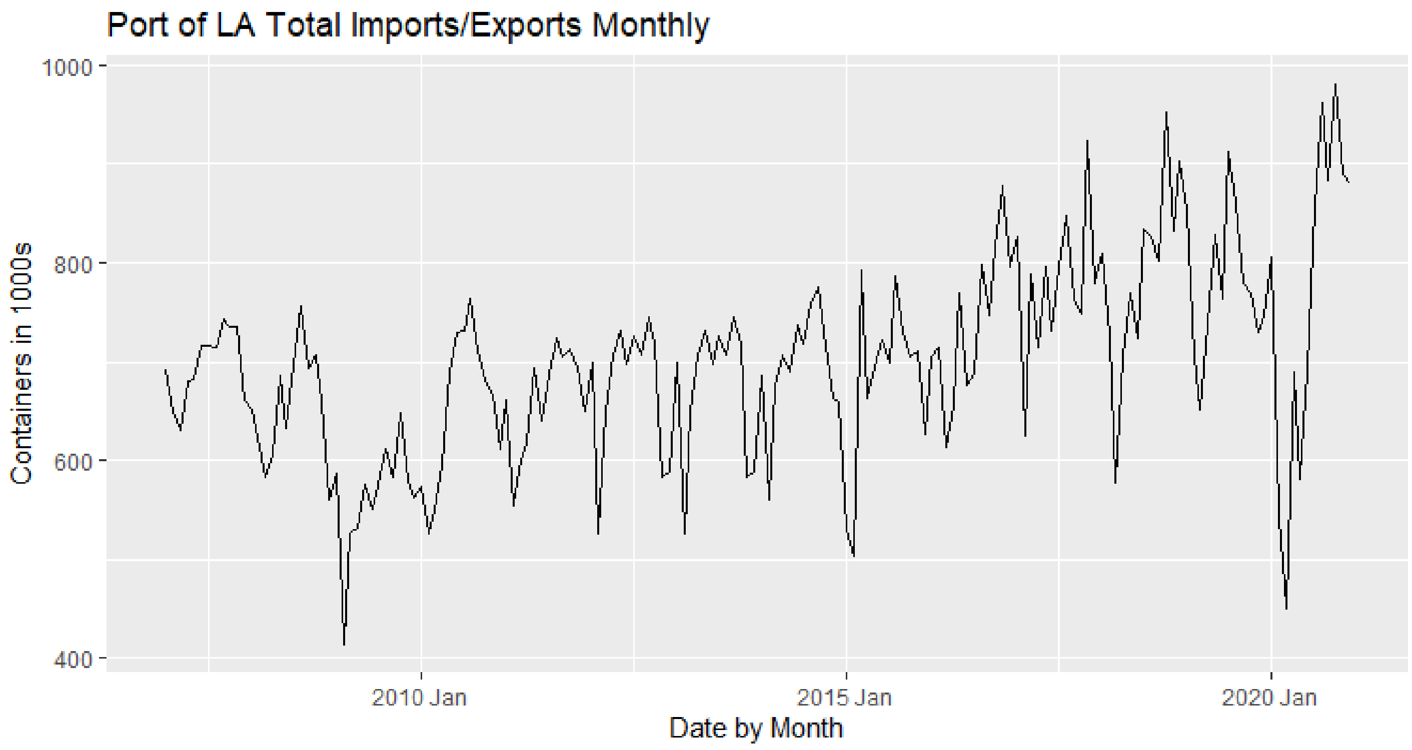
Appendix B: ETS Fit – Total TEUs

Testing ETS models:

Selected A, A, A to test first to determine if there was upward linear trend with additive seasonality
AIC = 2272.346

Selected A, M, A second to test for upward exponential trend with additive seasonality
AIC = 2272.471

Last Selected A, N, A to determine if there was a constant trend with additive seasonality
AIC = 2268.760



```
##Jason Rogers 669 Final Project##

##load libraries##

library(tsibble)
```

```
##
## Attaching package: 'tsibble'
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, union
```

```
library(dplyr)
```

```
## Warning: package 'dplyr' was built under R version 4.2.3
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(fpp3)
```

```
## Warning: package 'fpp3' was built under R version 4.2.3
```

```
## ── Attaching packages ──────────────────────────────────────────────── fpp3 0.5 ──
```

```
## ✓ tibble      3.2.1     ✓ tsibbledata 0.4.1
## ✓ tidyr       1.3.0     ✓ feasts      0.3.0
## ✓ lubridate   1.8.0     ✓ fable       0.3.2
## ✓ ggplot2     3.4.3     ✓ fabletools  0.3.2
```

```
## Warning: package 'tibble' was built under R version 4.2.3
```

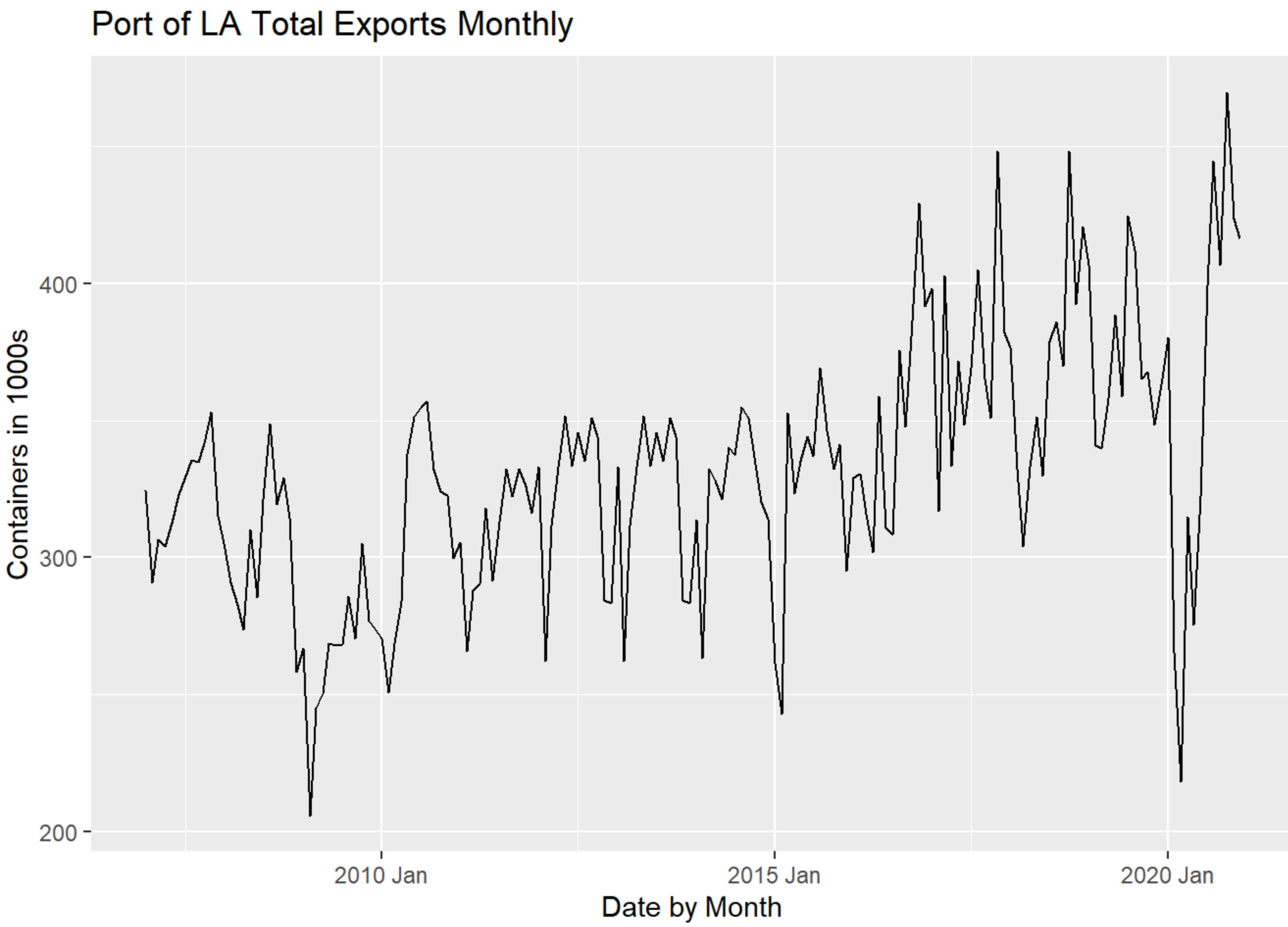
```
## Warning: package 'tidyr' was built under R version 4.2.3
```

```
## Warning: package 'ggplot2' was built under R version 4.2.3
```

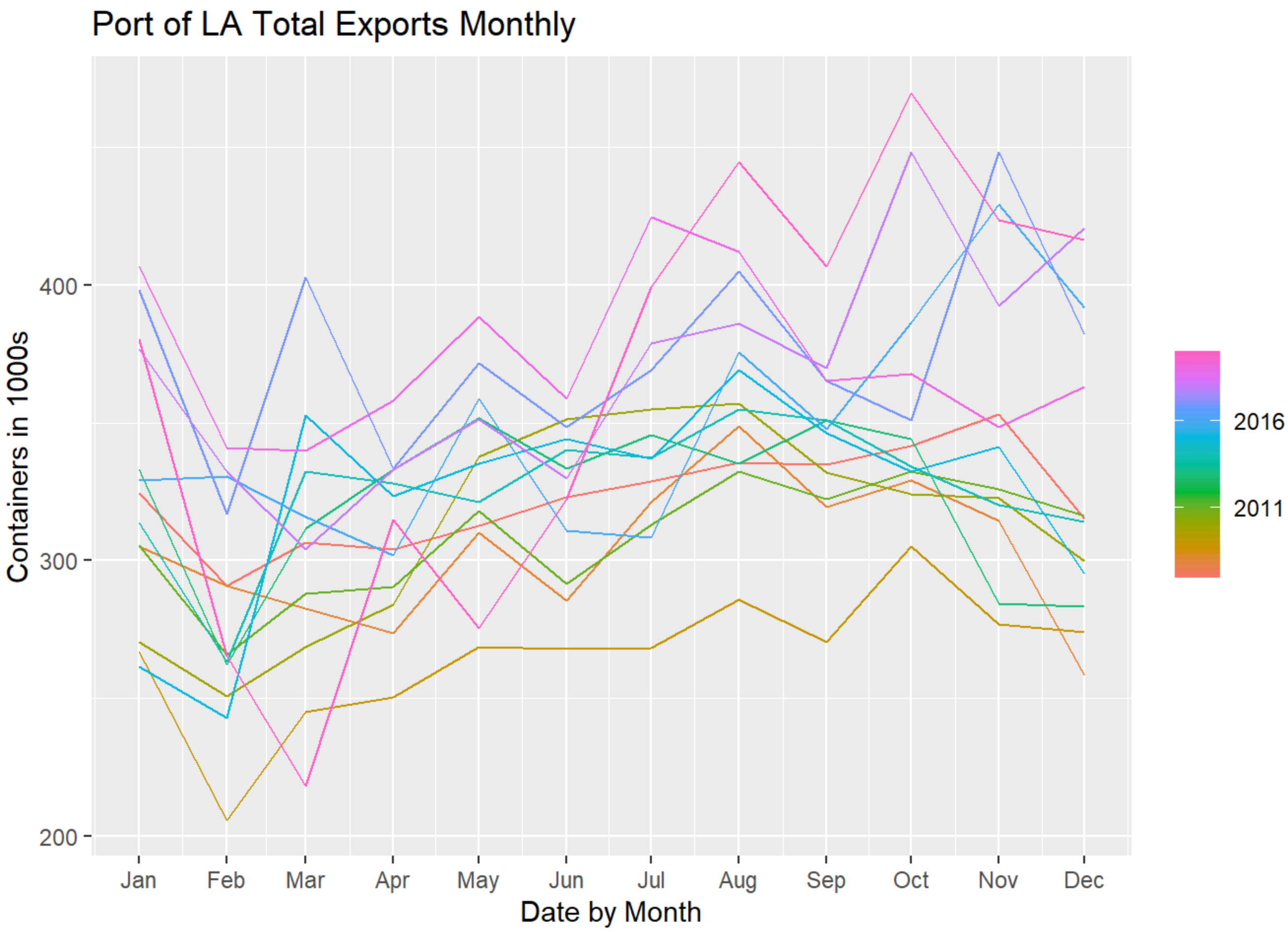
```
## ── Conflicts ──────────────────────────────────────────────────── fpp3_conflicts ──
## ✖ lubridate::date()     masks base::date()
## ✖ dplyr::filter()       masks stats::filter()
## ✖ tsibble::intersect()  masks base::intersect()
## ✖ lubridate::interval() masks tsibble::interval()
## ✖ dplyr::lag()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

```
##set directory##

setwd("C:/Users/roger/OneDrive/Desktop/SCMA 669 Merrick")

##Load dataset##

exports.df <- read.csv("Containers Port of Los Angeles Exports.csv", header = TRUE)

exports.df <- exports.df %>%
  select(Date, Exports)

View(exports.df)

##format Date column and make Month##

exports.tb <- exports.df %>%
  mutate(Month = as.Date(Date, "%m/%d/%Y")) %>%
  mutate(Date = yearmonth(Month))

##set index and create tsibble##

exports.tb <- exports.tb %>%
  as_tsibble(index = Date)

##partition the data for training##

train.tb <- exports.tb %>%
  filter(year(Date) <= 2020)

##Visualize your training data##
```

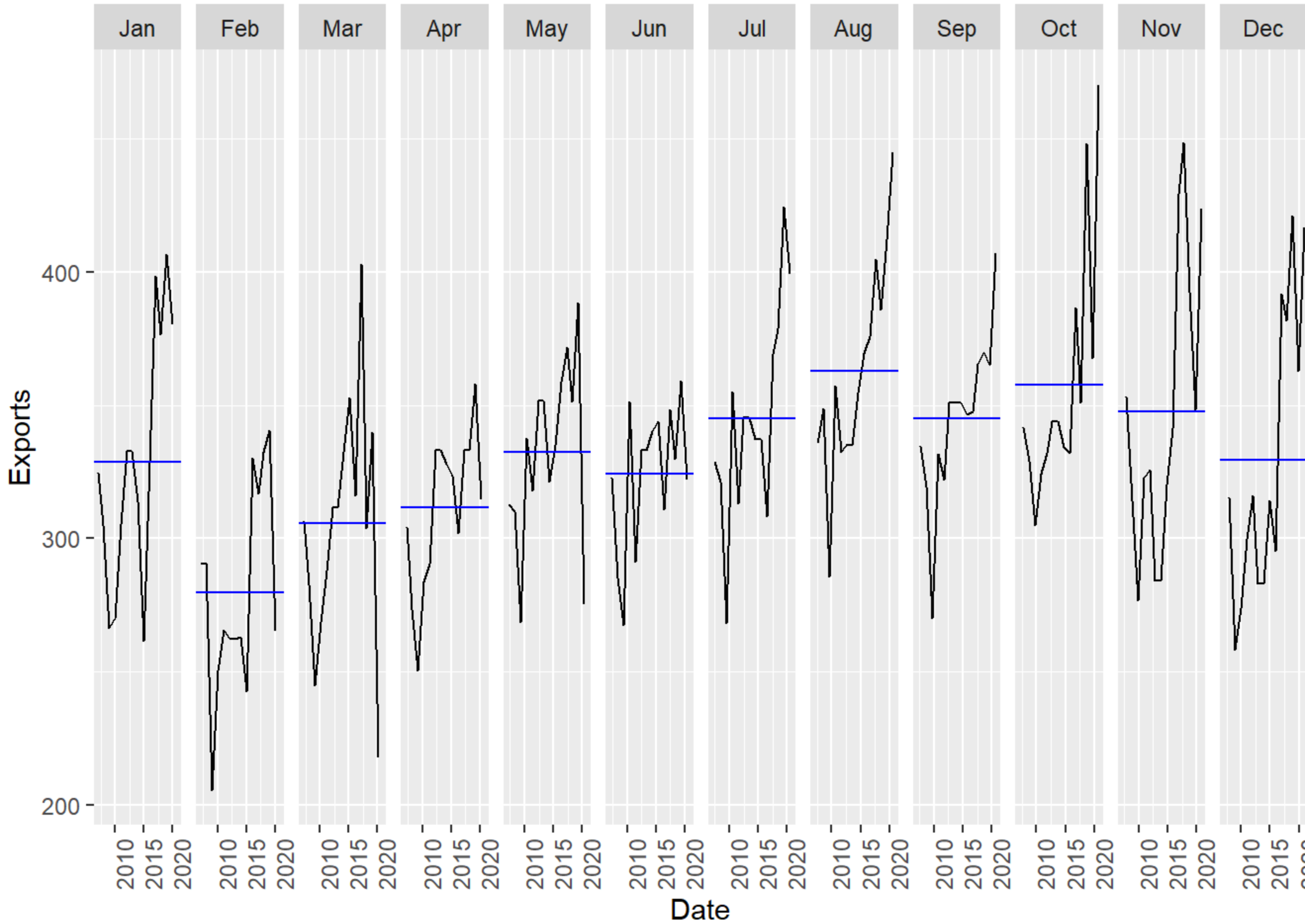
```
# Create a time series plot.
train.tb %>% autoplot(Exports) +
  ylab("Containers in 1000s") +
  xlab("Date by Month") +
  ggtitle("Port of LA Total Exports Monthly")
```




```
# Create a seasonal plot.
train.tb %>% gg_season(Exports, period = "year") +
  ylab("Containers in 1000s") +
  xlab("Date by Month") +
  ggtitle("Port of LA Total Exports Monthly")
```




```
# Create a sub-series plot.
train.tb %>% gg_subseries(Exports)
```

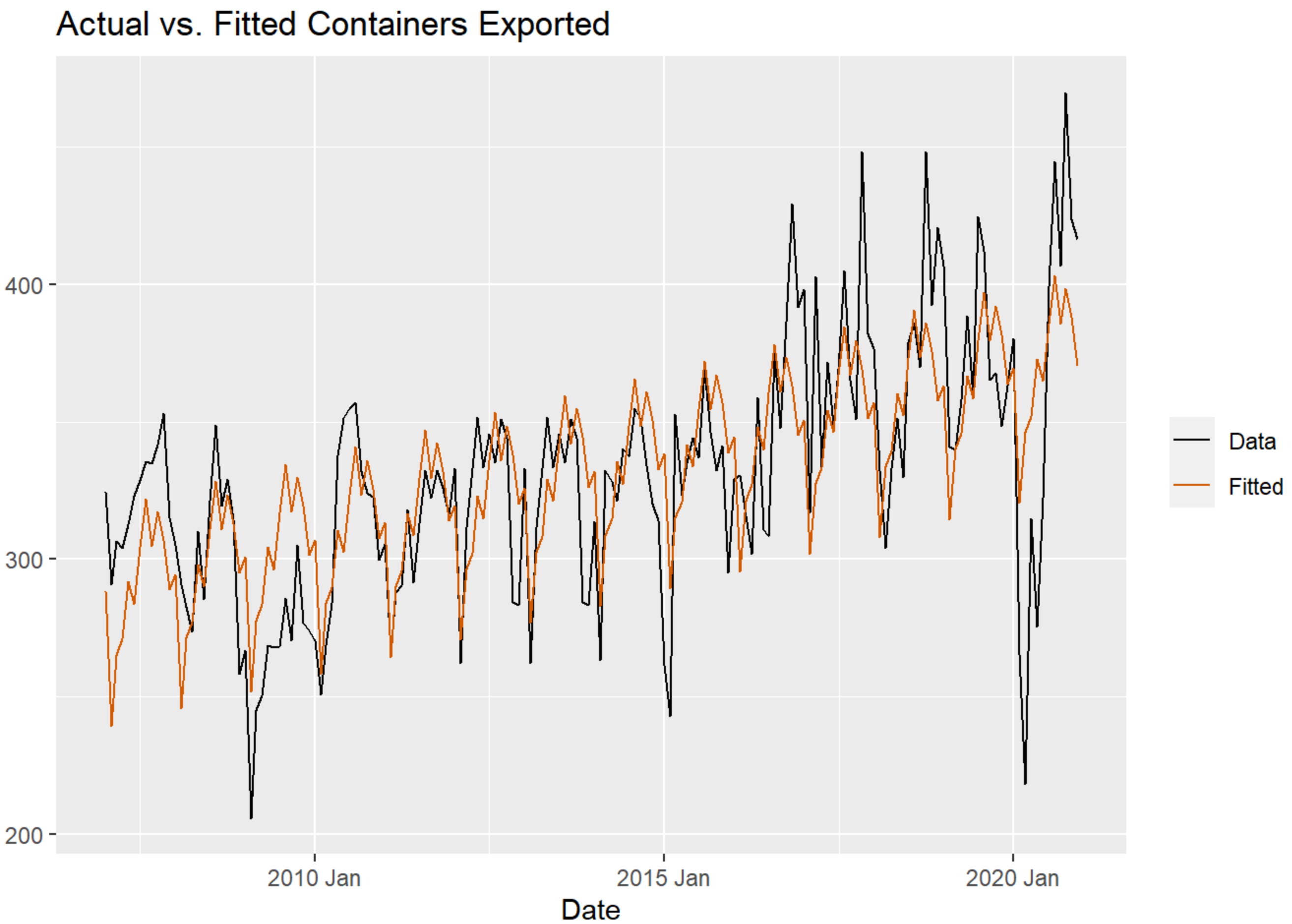



```
##There is a slight upward linear trend with seasonality that is additive. The trend has more ex
planatory power than the seasonality. The graph also shows how exports were impacted during COVI
D, both 2008 and 2020 break from the trend and dip heavily.##
```

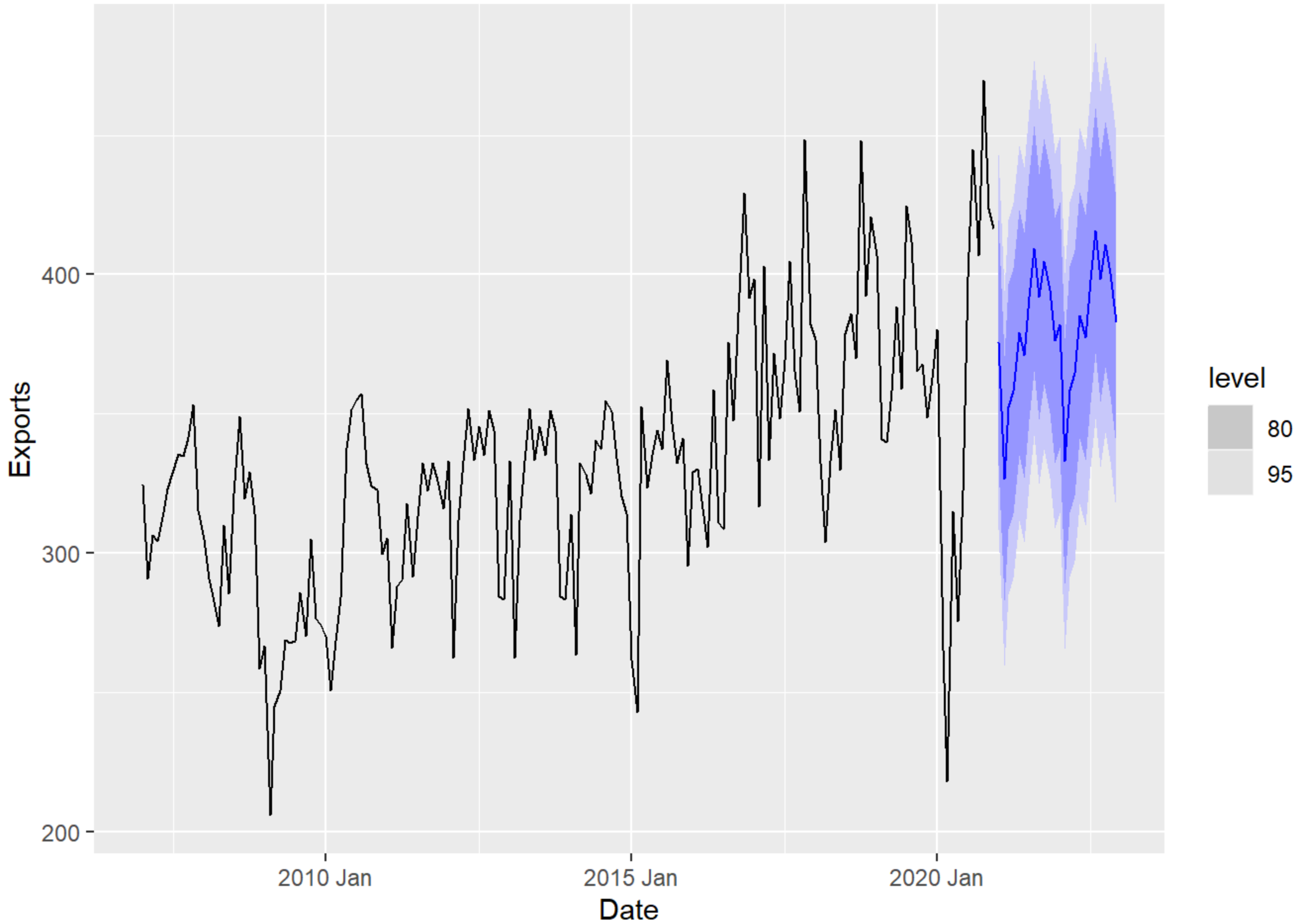
```
##Fit a regression model to the training data with appropriate trend and seasonality##

train.tb_lm <- train.tb %>%
  model(TSLM(Exports ~ trend() + season()))%>%
  report()
```

```
## Series: Exports
## Model: TSLM
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -128.3113  -17.5576    0.1169   19.8077   79.1221
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)     287.7620     9.7077  29.643  < 2e-16 ***
## trend()           0.5203     0.0524   9.929  < 2e-16 ***
## season()year2   -49.4339    12.4180  -3.981 0.000105 ***
## season()year3   -24.2170    12.4184  -1.950 0.052969 .
## season()year4   -18.8959    12.4189  -1.522 0.130161
## season()year5     1.3516    12.4197   0.109 0.913480
## season()year6    -7.1972    12.4207  -0.579 0.563126
## season()year7    13.1325    12.4219   1.057 0.292061
## season()year8    30.1480    12.4233   2.427 0.016383 *
## season()year9    12.1834    12.4250   0.981 0.328340
## season()year10   24.2597    12.4269   1.952 0.052716 .
## season()year11   13.3978    12.4290   1.078 0.282730
## season()year12   -5.0846    12.4313  -0.409 0.683090
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 32.85 on 155 degrees of freedom
## Multiple R-squared: 0.5336,  Adjusted R-squared: 0.4975
## F-statistic: 14.78 on 12 and 155 DF, p-value: < 2.22e-16
```

```
augment(train.tb_lm) %>%
  ggplot(aes(x = Date)) +
  geom_line(aes(y = Exports, colour = "Data")) +
  geom_line(aes(y = .fitted, colour = "Fitted")) +
  labs(y = NULL,
       title = "Actual vs. Fitted Containers Exported"
  ) +
  scale_colour_manual(values=c(Data="black",Fitted="#D55E00")) +
  guides(colour = guide_legend(title = NULL))
```




```
##Forecasting 24 months##

fc_train.tb_lm <- train.tb_lm %>%
  forecast(h = 24)

fc_train.tb_lm %>%
  autoplot(train.tb)
```




```
accuracy(fc_train.tb_lm, exports.tb)
```

```
## # A tibble: 1 × 10
##   .model                     .type    ME  RMSE   MAE   MPE  MAPE  MASE RMSSE  ACF1
##   <chr>                      <chr> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 TSLM(Exports ~ trend() … Test   37.3  62.2  54.7  7.89  13.0  1.78  1.52 0.776
```

```
##Mape 13##

##Fit an ARIMA model##

##remove any seasonal trend? (D)##

train.tb %>% features(Exports, unitroot_nsdiffs)
```

```
## # A tibble: 1 × 1
##   nsdiffs
##     <int>
## 1       0
```

```
##We do seasonal (lag 12) difference due to the results being 0. D=0##

##remove any non-seasonal trend? (d)##

train.tb %>% features(Exports, unitroot_ndiffs)
```

```
## # A tibble: 1 × 1
##   ndiffs
##    <int>
## 1      1
```

```
##The number 12 due to monthly data for 1 LAG, d = 1.##

library(gridExtra)
```

```
##
## Attaching package: 'gridExtra'
```

```
## The following object is masked from 'package:dplyr':
##
##     combine
```

```
grid.arrange(train.tb %>% autoplot(Exports),
             train.tb %>% autoplot(difference(Exports, 1)),
             train.tb %>% autoplot(difference(Exports, 2)),
             train.tb %>% autoplot(difference(Exports, 12)),
             train.tb %>% autoplot(difference(difference(Exports, 1), 12)),
             train.tb %>% autoplot(difference(difference(Exports, 2), 12)),
             ncol = 3, nrow = 2)
```

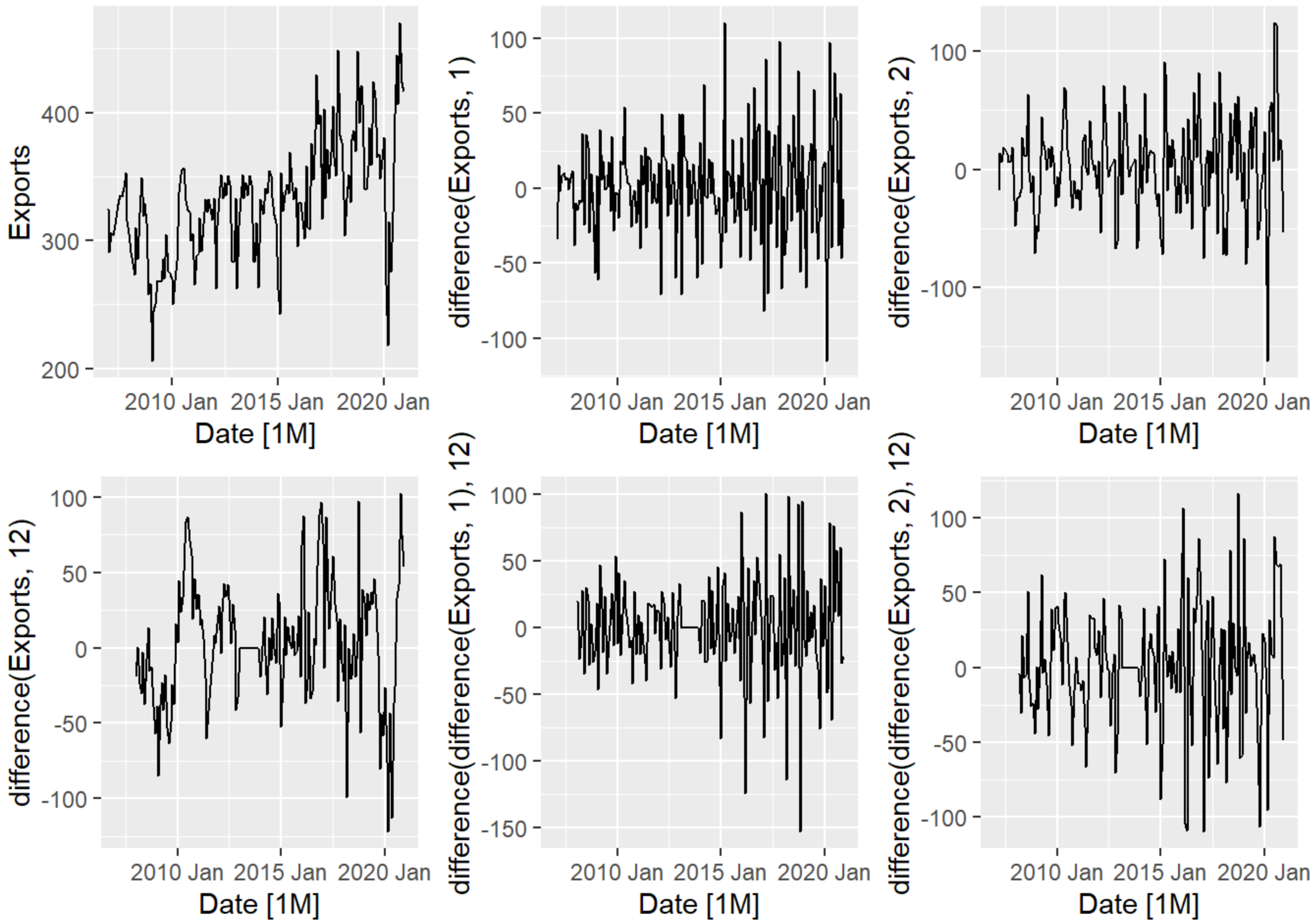
```
## Warning: Removed 1 row containing missing values (`geom_line()`).
```

```
## Warning: Removed 2 rows containing missing values (`geom_line()`).
```

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

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

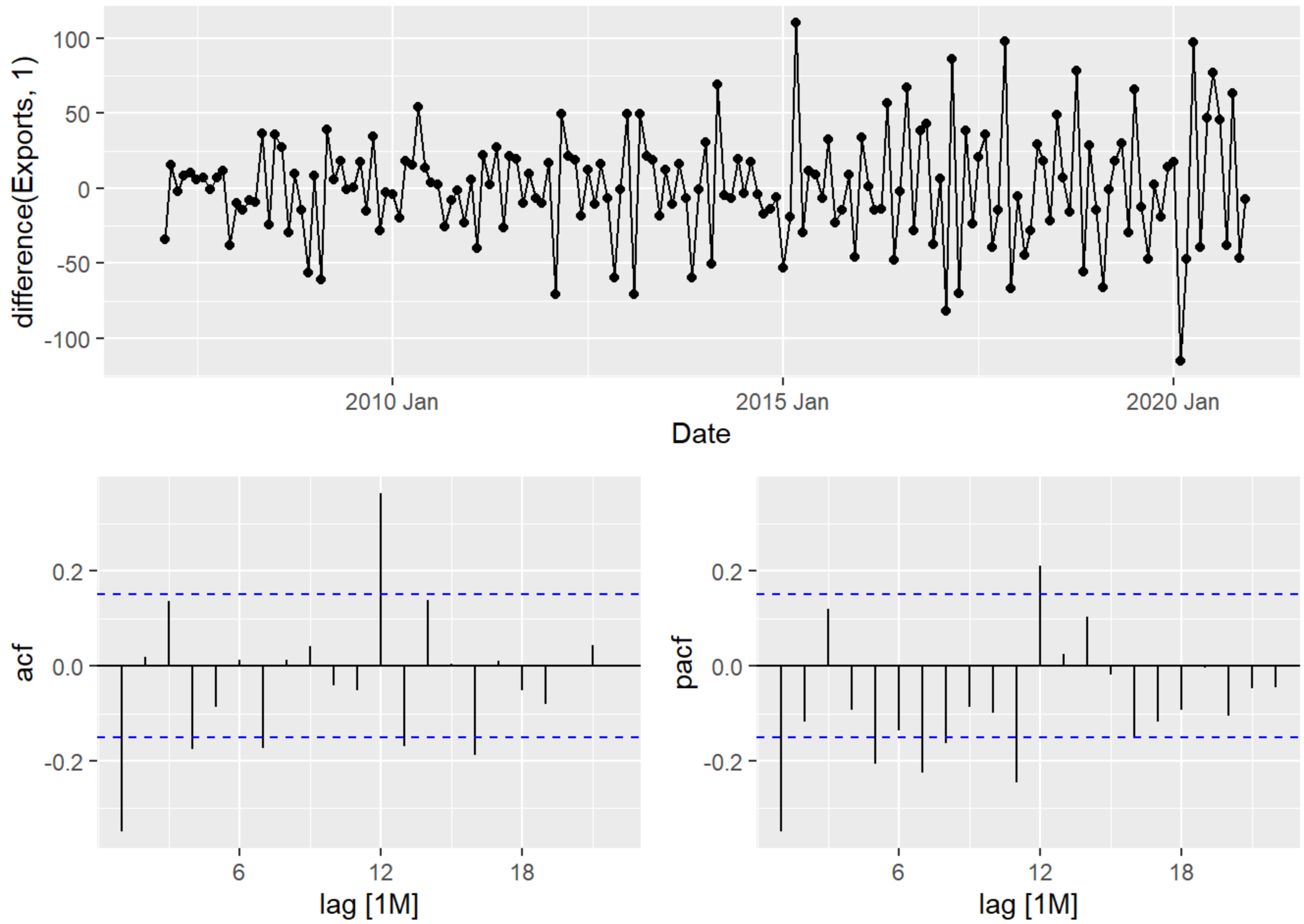
```
## Warning: Removed 14 rows containing missing values (`geom_line()`).
```




```
##ACF and PACF##

train.tb %>% gg_tsdisplay(difference(Exports, 1), plot_type='partial')
```

```
## Warning: Removed 1 row containing missing values (`geom_line()`).
```

```
## Warning: Removed 1 rows containing missing values (`geom_point()`).
```




```
# Graph shows spikes at AR1, AR5 and AR7 without a solid pattern forming in ACF.
# There is a spike at AR12 to consider

##Let's try d = 1, D = 0, p = 0, P = 1##

ARIMA1.fit <- train.tb %>% model(ARIMA(Exports ~ pdq(0,1,0) + PDQ(1,0,0)))
report(ARIMA1.fit)
```

```
## Series: Exports
## Model: ARIMA(0,1,0)(1,0,0)[12]
##
## Coefficients:
##          sar1
##        0.4253
## s.e.   0.0750
##
## sigma^2 estimated as 1100:  log likelihood=-822.44
## AIC=1648.88   AICc=1648.95   BIC=1655.11
```

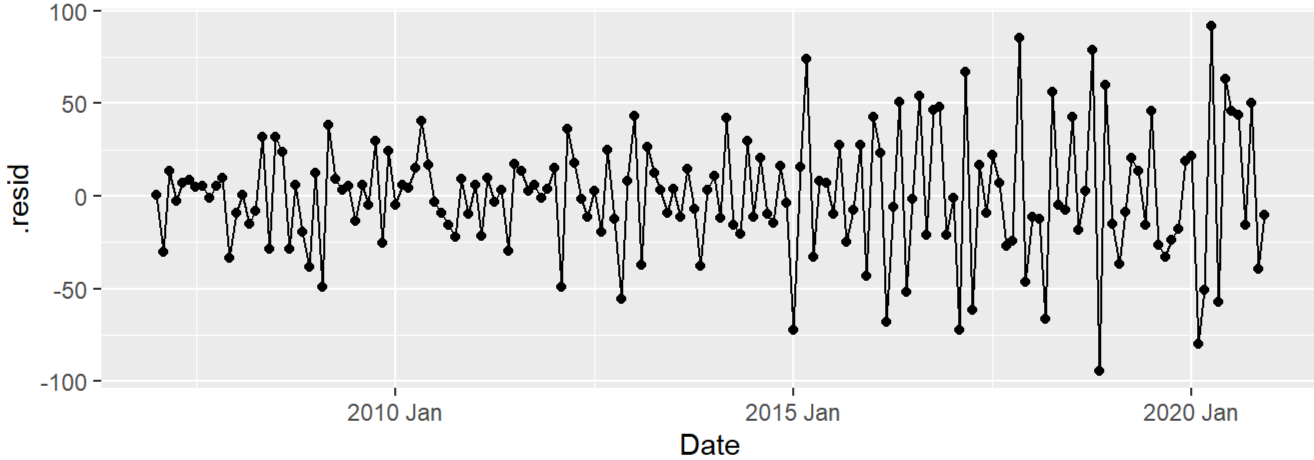
```
##AIC 1648.88##

ARIMA2.fit <- train.tb %>% model(ARIMA(Exports ~ 1 + pdq(0,1,0) + PDQ(0,0,1)))
report(ARIMA2.fit)
```

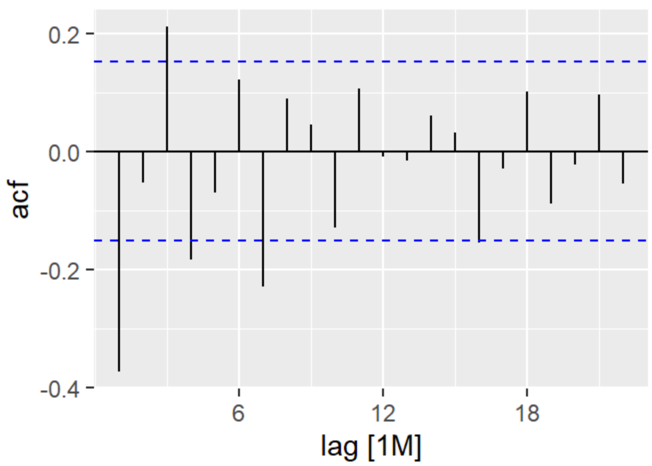
```
## Series: Exports
## Model: ARIMA(0,1,0)(0,0,1)[12] w/ drift
##
## Coefficients:
##         sma1  constant
##       0.2630    0.6303
## s.e.  0.0625    3.3146
##
## sigma^2 estimated as 1199:  log likelihood=-828.34
## AIC=1662.67   AICc=1662.82   BIC=1672.02
```

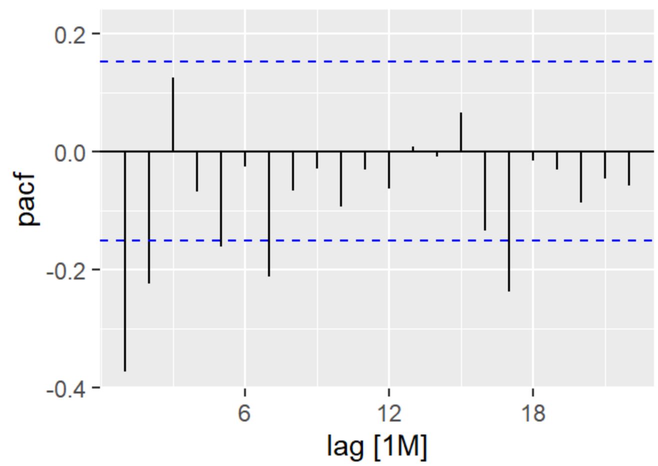
```
##AIC 1662.67##

ARIMA3.fit <- train.tb %>% model(ARIMA(Exports ~ 1 + pdq(0,1,0) + PDQ(1,0,1)))
report(ARIMA3.fit)
```

```
## Series: Exports
## Model: ARIMA(0,1,0)(1,0,1)[12] w/ drift
##
## Coefficients:
##         sar1     sma1  constant
##       0.7894  -0.4505    0.0915
## s.e.  0.1070   0.1632    1.1449
##
## sigma^2 estimated as 1049:  log likelihood=-818.17
## AIC=1644.33   AICc=1644.58   BIC=1656.81
```

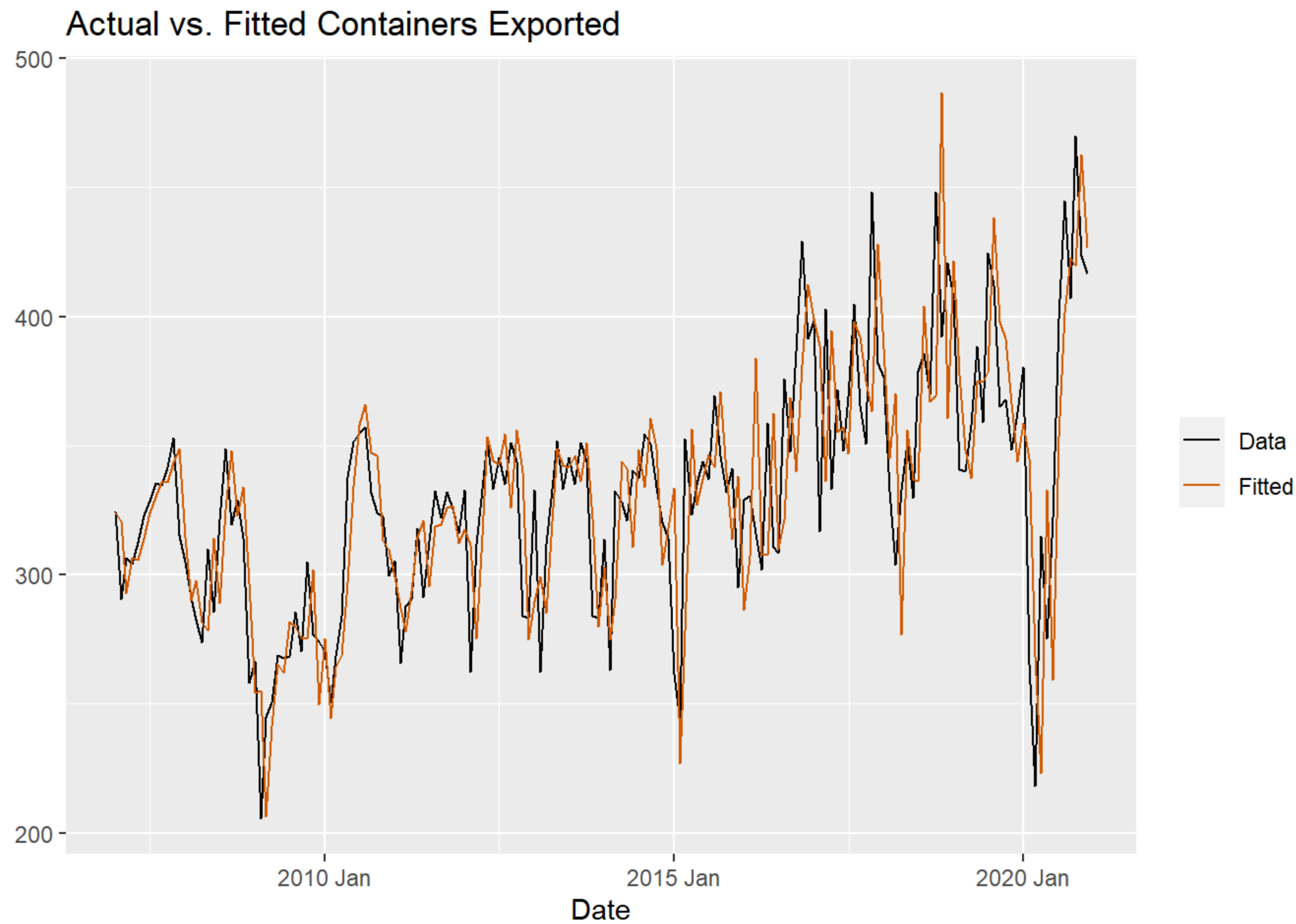
```
##AIC 1644.33, this has the lowest mape##

##Residuals##

augment(ARIMA3.fit) %>% gg_tsdisplay(.resid, plot_type='partial')
```



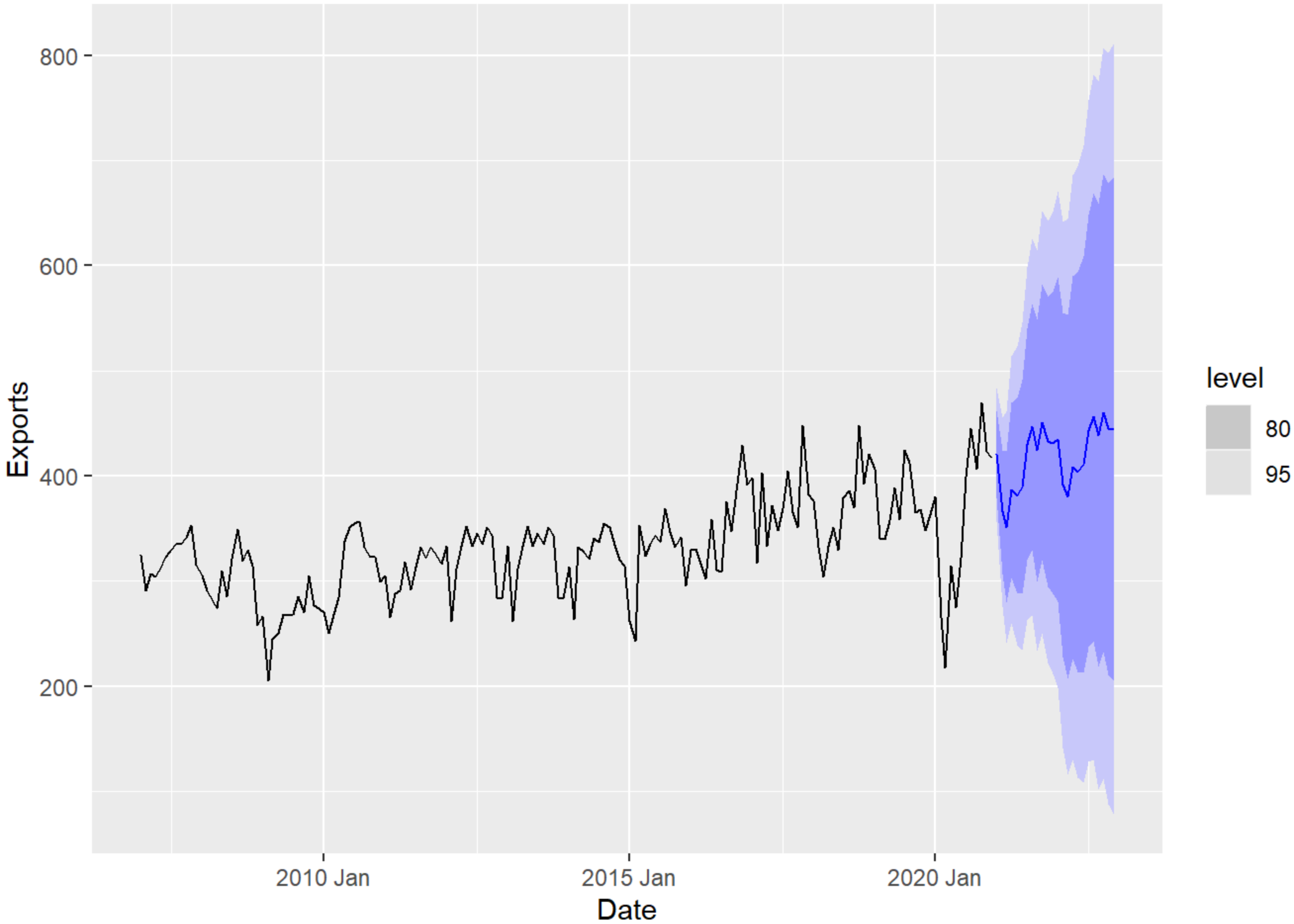





```
##Pattern on the left##

augment(ARIMA3.fit) %>%
  ggplot(aes(x = Date)) +
  geom_line(aes(y = Exports, colour = "Data")) +
  geom_line(aes(y = .fitted, colour = "Fitted")) +
  labs(y = NULL,
       title = "Actual vs. Fitted Containers Exported"
  ) +
  scale_colour_manual(values=c(Data="black",Fitted="#D55E00")) +
  guides(colour = guide_legend(title = NULL))
```




```
fc_ARIMA3.fit <- ARIMA3.fit %>%
  forecast(h = 24)

fc_ARIMA3.fit %>%
  autoplot(train.tb)
```



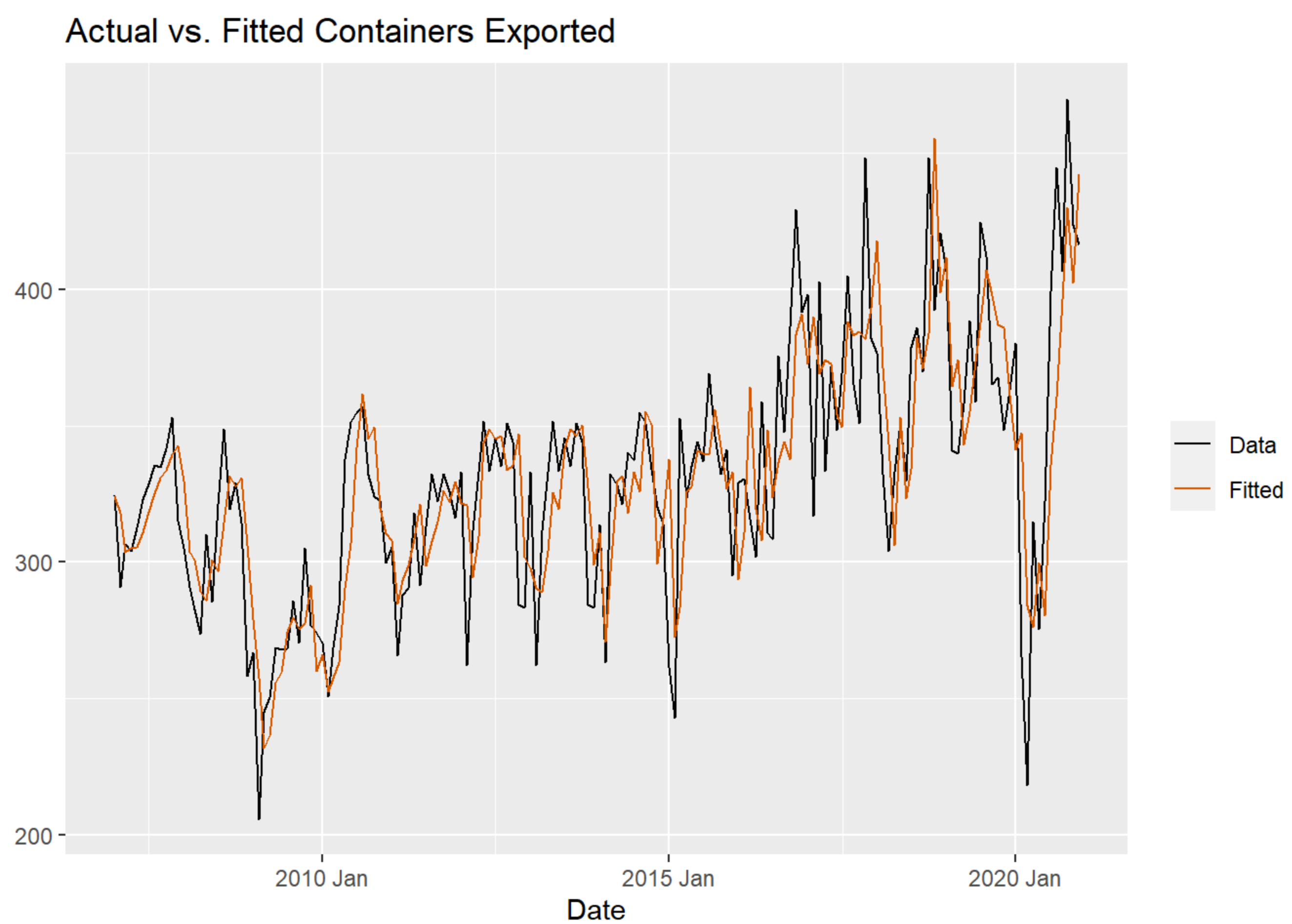

```
accuracy(fc_ARIMA3.fit, exports.tb)
```

```
## # A tibble: 1 × 10
##   .model                   .type    ME  RMSE   MAE   MPE  MAPE  MASE RMSSE  ACF1
##   <chr>                    <chr> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 ARIMA(Exports ~ 1 + pdq… Test  0.223  59.4  46.3 -1.25  11.5  1.50  1.45 0.723
```

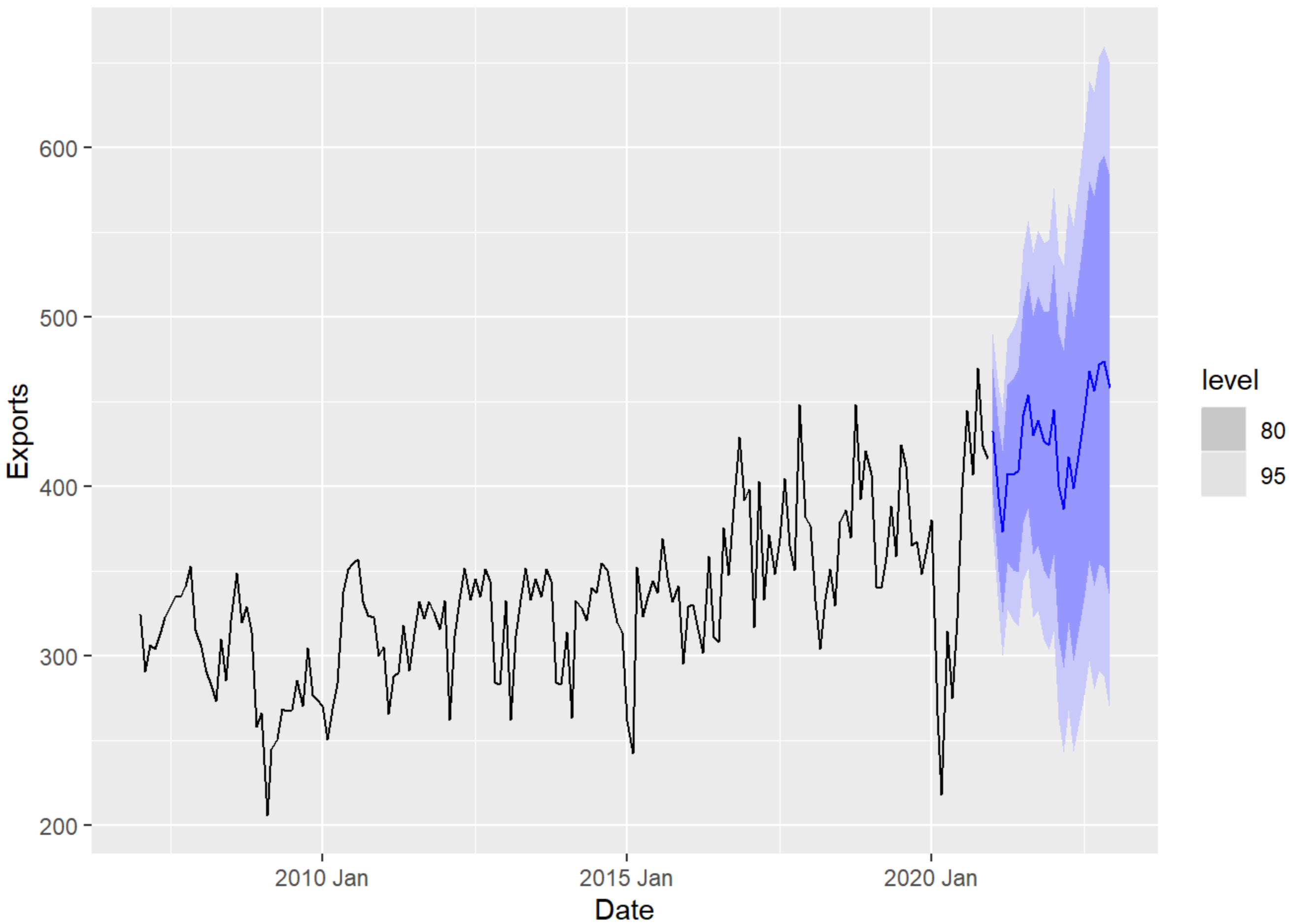
```
##Mape 11.5##

##auto.arima()##

ARIMA.auto <- train.tb %>% model(ARIMA(Exports))
report(ARIMA.auto)
```

```
## Series: Exports
## Model: ARIMA(0,1,1)(0,0,2)[12]
##
## Coefficients:
##           ma1    sma1    sma2
##       -0.4391  0.2986  0.4526
## s.e.   0.0759  0.0793  0.0850
##
## sigma^2 estimated as 854.8:  log likelihood=-802.26
## AIC=1612.52   AICc=1612.77   BIC=1624.99
```

```
##AIC 1612.52##

augment(ARIMA.auto) %>%
  ggplot(aes(x = Date)) +
  geom_line(aes(y = Exports, colour = "Data")) +
  geom_line(aes(y = .fitted, colour = "Fitted")) +
  labs(y = NULL,
       title = "Actual vs. Fitted Containers Exported"
  ) +
  scale_colour_manual(values=c(Data="black",Fitted="#D55E00")) +
  guides(colour = guide_legend(title = NULL))
```



```
fc_ARIMA.auto <- ARIMA.auto %>%
  forecast(h = 24)

fc_ARIMA.auto %>%
  autoplot(train.tb)
```




```
accuracy(fc_ARIMA.auto, exports.tb)
```

```
## # A tibble: 1 × 10
##   .model         .type    ME  RMSE   MAE   MPE  MAPE  MASE RMSSE  ACF1
##   <chr>          <chr> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 ARIMA(Exports) Test  -10.3  60.9  44.3 -3.87  11.3  1.44  1.49 0.704
```

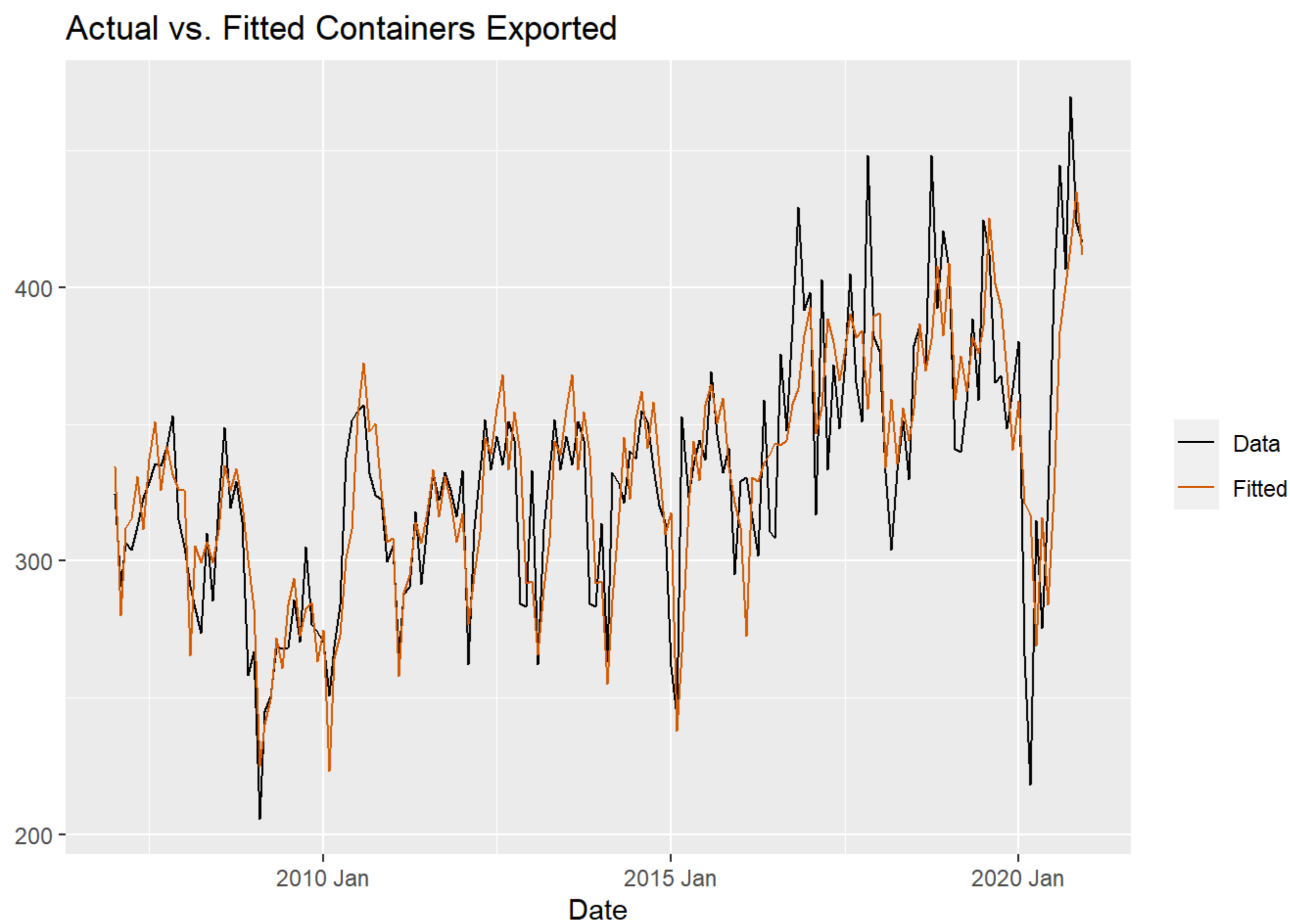
```
##Mape 11.3##

##exponential smoothing model##

##The ETS model I would first try would be a model with an additive linear trend with seasonalit
y##

ets_manual1 <- train.tb %>% model(ETS(Exports ~ error("A") + trend("A") + season("A")))
report(ets_manual1)
```

```
## Series: Exports
## Model: ETS(A,A,A)
##   Smoothing parameters:
##     alpha = 0.5482453
##     beta  = 0.0001000069
##     gamma = 0.0001100631
##
##   Initial states:
##      l[0]      b[0]      s[0]    s[-1]  s[-2]    s[-3]    s[-4]    s[-5]
##  332.9468 0.5518572 -4.040492 13.54669 24.187 13.42605 30.61627 12.91895
##      s[-6]    s[-7]     s[-8]     s[-9]    s[-10]    s[-11]
##  -6.102845 3.922654 -17.30098 -23.21312 -48.79914 0.8389554
##
##   sigma^2:  807.623
##
##      AIC     AICc      BIC
## 2002.620 2006.700 2055.727
```

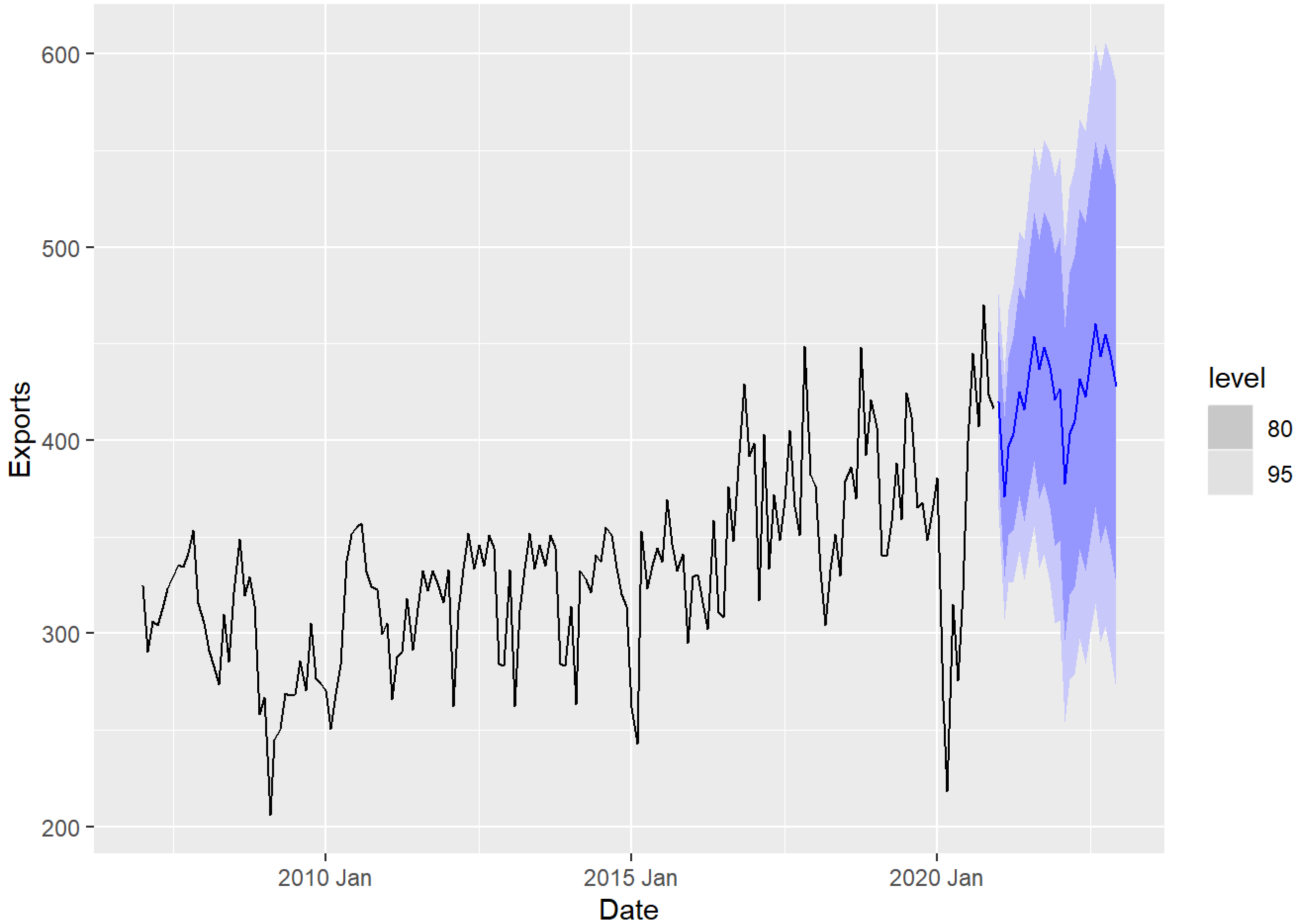
```
##AIC 2002.62

ets_manual2 <- train.tb %>% model(ETS(Exports ~ error("A") + trend("A") + season("M")))
report(ets_manual2)
```

```
## Series: Exports
## Model: ETS(A,A,M)
##   Smoothing parameters:
##     alpha = 0.4474613
##     beta  = 0.0001000051
##     gamma = 0.1698272
##
##   Initial states:
##      l[0]      b[0]      s[0]    s[-1]   s[-2]    s[-3]    s[-4]    s[-5]
##  331.6285 0.5417158 0.9611465 1.011077 1.04821 1.037531 1.055291 1.062521
##     s[-6]    s[-7]     s[-8]     s[-9]    s[-10]    s[-11]
##  1.016476 1.041335 0.9808745 0.9844664 0.8599736 0.9410976
##
##   sigma^2:  910.8587
##
##      AIC     AICc      BIC
## 2022.829 2026.909 2075.936
```

```
##Multiplicative does not perform better, we stick with manual1##

augment(ets_manual1) %>%
  ggplot(aes(x = Date)) +
  geom_line(aes(y = Exports, colour = "Data")) +
  geom_line(aes(y = .fitted, colour = "Fitted")) +
  labs(y = NULL,
       title = "Actual vs. Fitted Containers Exported"
  ) +
  scale_colour_manual(values=c(Data="black",Fitted="#D55E00")) +
  guides(colour = guide_legend(title = NULL))
```




```
fc_ets_manual1 <- ets_manual1 %>%
  forecast(h = 24)

fc_ets_manual1 %>%
  autoplot(train.tb)
```



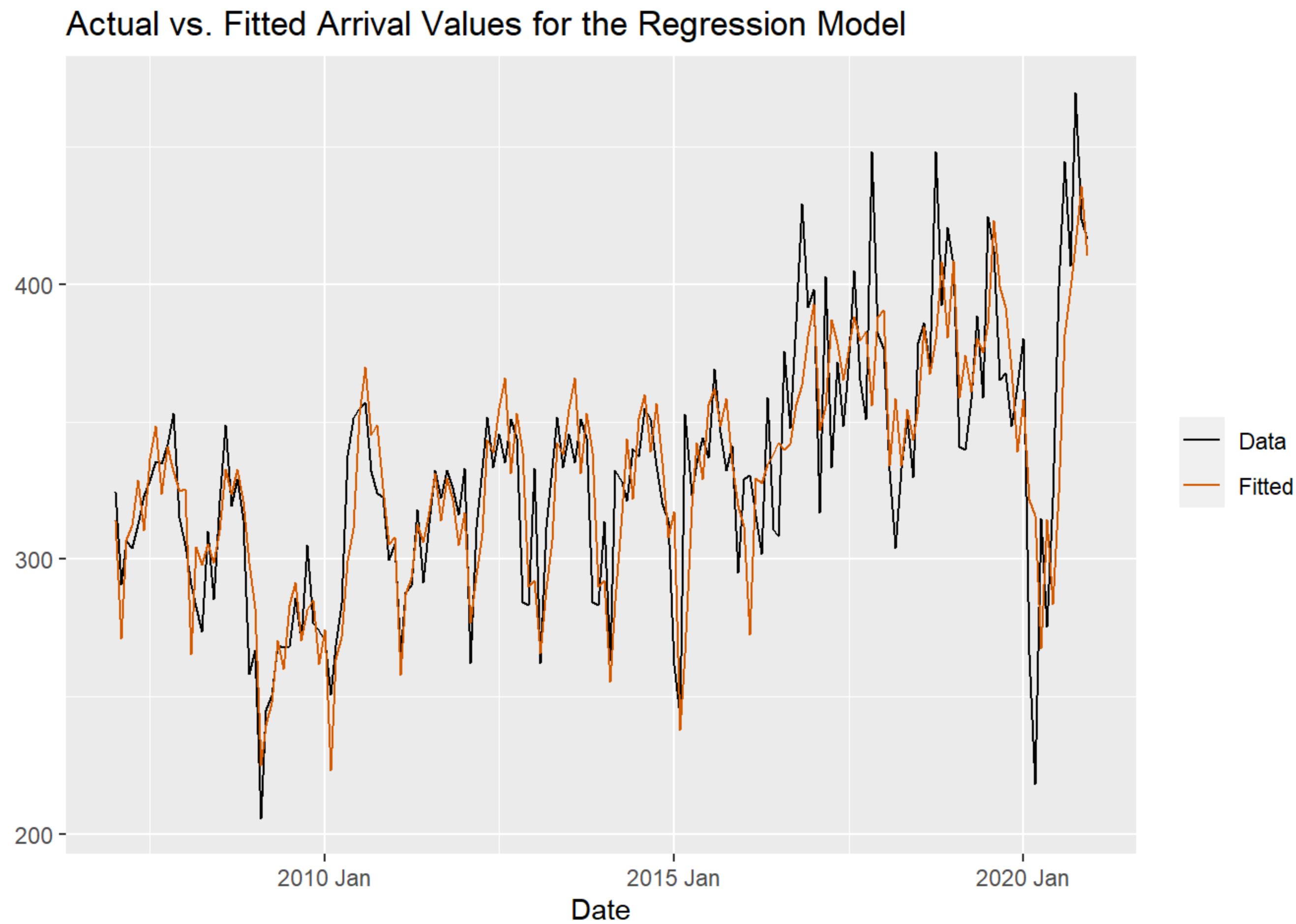

```
accuracy(fc_ets_manual1, exports.tb)
```

```
## # A tibble: 1 × 10
##   .model                    .type    ME  RMSE   MAE   MPE  MAPE  MASE RMSSE  ACF1
##   <chr>                     <chr> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 "ETS(Exports ~ error(\"… Test  -7.27  50.2  38.3 -2.86  9.75  1.25  1.23 0.777
```

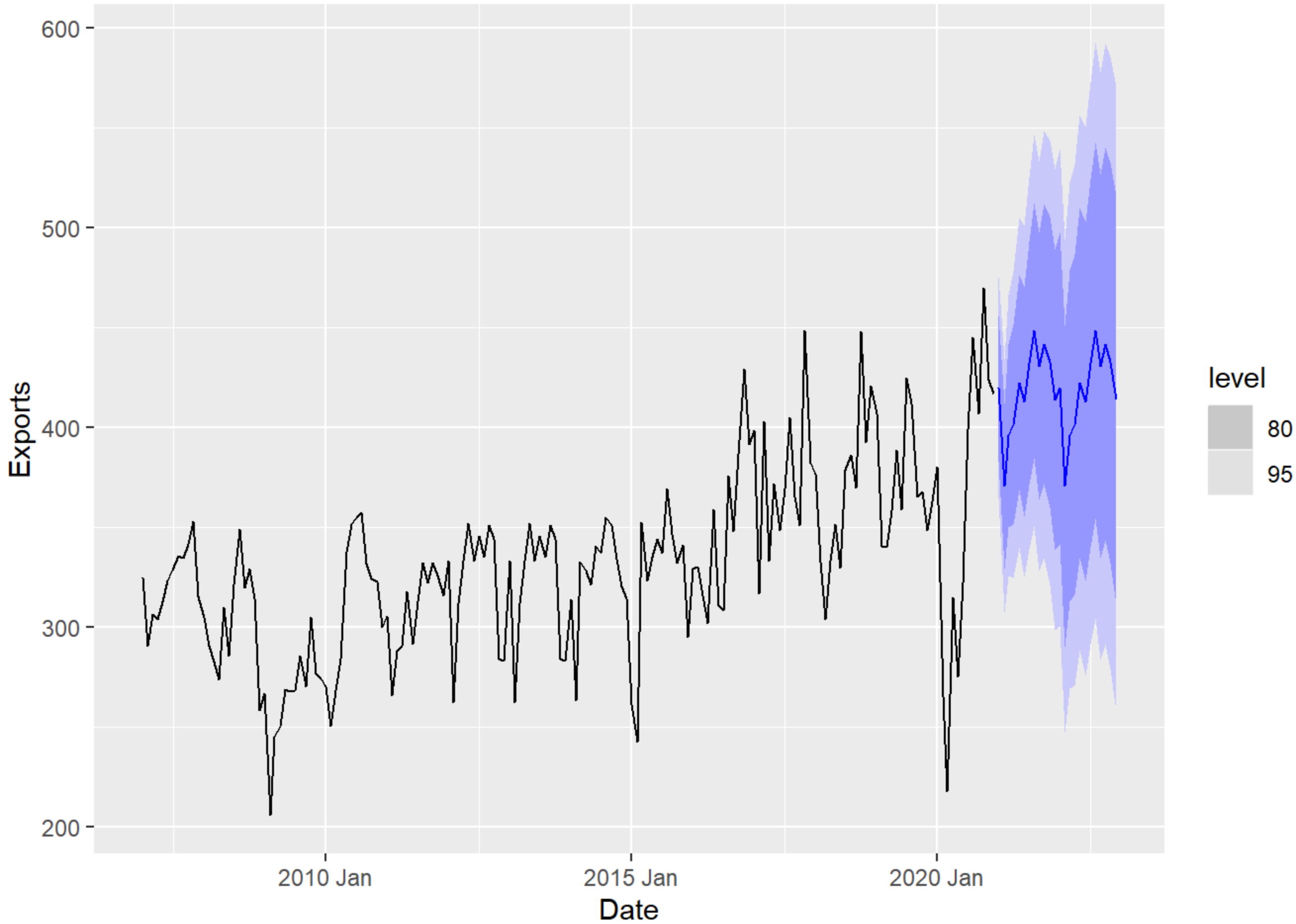
```
##Mape 9.75##

##Auto ETS##

ets_auto <- train.tb %>% model(ETS(Exports))
report(ets_auto)
```

```
## Series: Exports
## Model: ETS(A,N,A)
##   Smoothing parameters:
##     alpha = 0.550009
##     gamma = 0.0001000379
##
##   Initial states:
##     l[0]      s[0]    s[-1]    s[-2]    s[-3]    s[-4]    s[-5]     s[-6]
##  313.051 -4.749498 13.93889 23.21871 12.14124 29.94124 13.66059 -5.491295
##     s[-7]     s[-8]     s[-9]    s[-10]   s[-11]
##  3.895073 -17.10702 -22.65474 -47.82946 1.036267
##
##   sigma^2:  799.458
##
##      AIC     AICc      BIC
## 1999.109 2002.267 2045.968
```

```
##AIC 1999.11##

augment(ets_auto) %>%
  ggplot(aes(x = Date)) +
  geom_line(aes(y = Exports, colour = "Data")) +
  geom_line(aes(y = .fitted, colour = "Fitted")) +
  labs(y = NULL,
       title = "Actual vs. Fitted Arrival Values for the Regression Model"
  ) +
  scale_colour_manual(values=c(Data="black",Fitted="#D55E00")) +
  guides(colour = guide_legend(title = NULL))
```




```
fc_ets_auto <- ets_auto %>%
  forecast(h = 24)

fc_ets_auto %>%
  autoplot(train.tb)
```



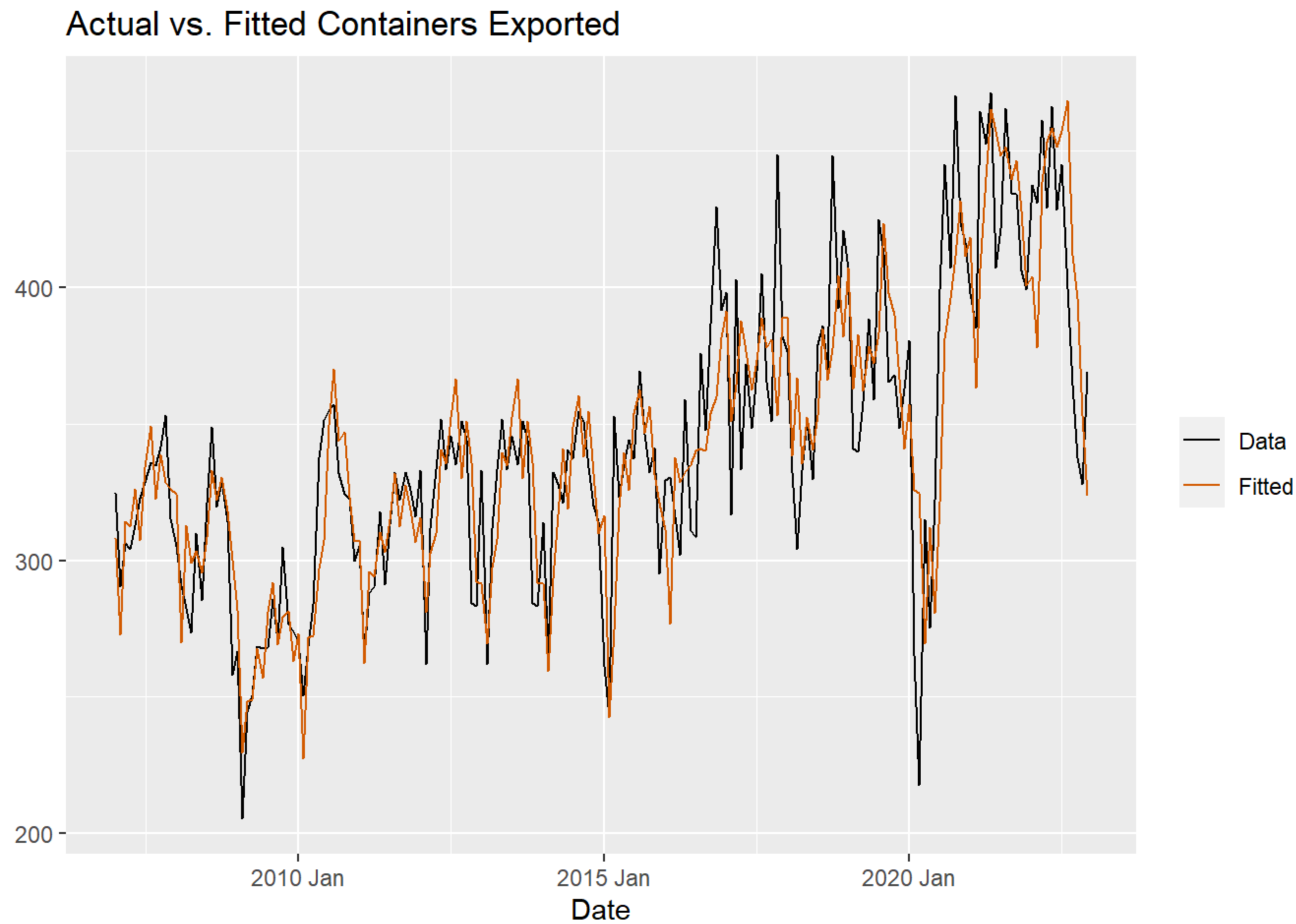

```
accuracy(fc_ets_auto, exports.tb)
```

```
## # A tibble: 1 × 10
##   .model       .type     ME  RMSE   MAE   MPE  MAPE  MASE RMSSE  ACF1
##   <chr>        <chr>  <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 ETS(Exports) Test  -0.437  47.2  37.6 -1.16  9.40  1.22  1.16 0.757
```

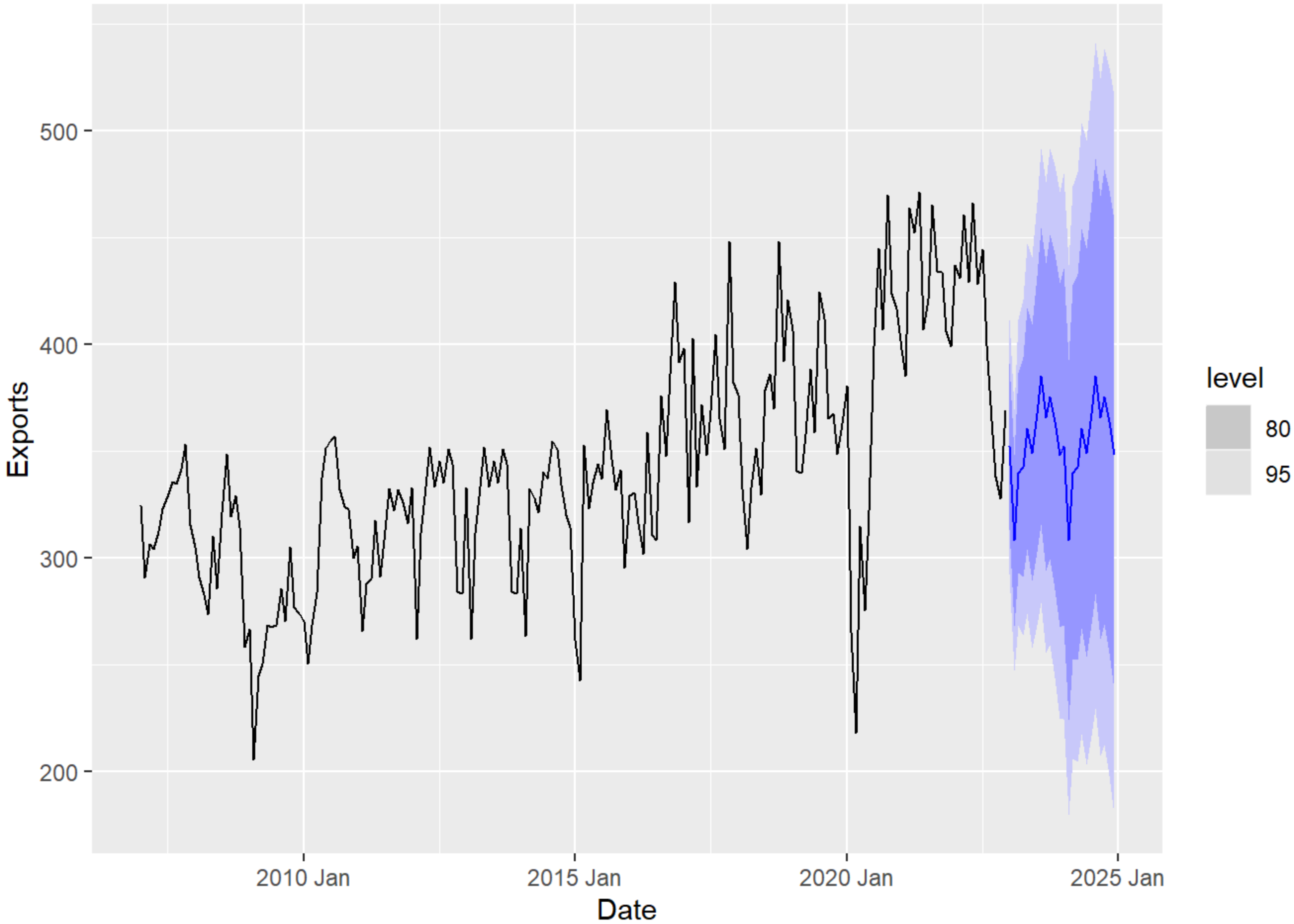
```
##Mape 9.40##

##Ensemble##

##Predictive accuracy of your five models in cross-validation##

all.models.fit <- train.tb %>%
  model(train.tb_lm = TSLM(Exports ~ trend() + season()),
        ARIMA3.fit = ARIMA(Exports ~ 1 + pdq(0,1,0) + PDQ(1,0,1)),
        ARIMA.auto = ARIMA(Exports),
        ets_manual1 = ETS(Exports ~ error("A") + trend("A") + season("A")),
        ets_auto = ETS(Exports),
        combination = combination_model(ARIMA(Exports ~ 1 + pdq(0,1,0) + PDQ(1,0,1)),
                                        ARIMA(Exports),
                                        TSLM(Exports ~ trend() + season()),
                                        ETS(Exports ~ error("A") + trend("A") + season("A")),
                                        ETS(Exports),
                                        cmbn_args = list(weights = "equal")))

all.models.pred <- all.models.fit %>% forecast(h = 24)

all.models.pred %>% accuracy(exports.tb) %>% arrange(MAPE, decreasing = TRUE)
```

```
## # A tibble: 6 × 10
##   .model      .type      ME  RMSE   MAE    MPE  MAPE  MASE RMSSE  ACF1
##   <chr>       <chr>   <dbl> <dbl> <dbl>  <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 ets_auto    Test   -0.437  47.2  37.6 -1.16   9.40  1.22  1.16 0.757
## 2 ets_manual1 Test   -7.27   50.2  38.3 -2.86   9.75  1.25  1.23 0.777
## 3 combination Test    3.89   53.0  41.7 -0.249 10.4   1.36  1.30 0.753
## 4 ARIMA.auto  Test  -10.3    60.9  44.3 -3.87  11.3   1.44  1.49 0.704
## 5 ARIMA3.fit  Test    0.223  59.4  46.3 -1.25  11.5   1.50  1.45 0.723
## 6 train.tb_lm Test   37.3    62.2  54.7  7.89  13.0   1.78  1.52 0.776
```

```
##Auto ETS performs best, Mape of 9.40##

##Fit recommended model to validation data.##

ets_auto_full <- exports.tb %>% model(ETS(Exports))

augment(ets_auto_full) %>%
  ggplot(aes(x = Date)) +
  geom_line(aes(y = Exports, colour = "Data")) +
  geom_line(aes(y = .fitted, colour = "Fitted")) +
  labs(y = NULL,
       title = "Actual vs. Fitted Containers Exported"
  ) +
  scale_colour_manual(values=c(Data="black",Fitted="#D55E00")) +
  guides(colour = guide_legend(title = NULL))
```




```
fc_ets_auto_full <- ets_auto_full %>%
  forecast(h = 24)

fc_ets_auto_full %>%
  autoplot(exports.tb)
```


500
400
300
200
Exports
2010 Jan
2015 Jan
2020 Jan
2025 Jan
Date
level
80
95