dynamic regression attempt

Daniel L.

5/13/2022

Auto-fit model

https://otexts.com/fpp3/regarima.html

The function ARIMA() will fit a regression model with ARIMA errors if exogenous regressors are included in the formula.

```
auto_fit <- nyc_ts_2 %>%
  model(ARIMA(tw_diff1 ~ cpi_diff1 + unemp_diff1 + avg_precip_diff1 + temp_diff1 + cdd_diff1))

report(auto_fit)
```

```
## Series: tw_diff1
## Model: LM w/ ARIMA(5,0,0) errors
##
## Coefficients:
##           ar1      ar2      ar3      ar4      ar5  cpi_diff1  unemp_diff1
##       -0.9083  -0.7830  -0.7243  -0.7495  -0.4124   -95.8412     96231.25
## s.e.   0.0665   0.0776   0.0804   0.0791   0.0679   140.9419     48970.48
##       avg_precip_diff1  temp_diff1  cdd_diff1
##               26793.86    929.3943  -861.8094
## s.e.          10723.35    149.9121   536.5374
##
## sigma^2 estimated as 1.89e+08:  log likelihood=-2087.54
## AIC=4197.07   AICc=4198.54   BIC=4232.91
```

The model returns

Auto-fit model with constant

```
auto_fit_constant <- nyc_ts_2 %>%
  model(ARIMA(tw_diff1 ~ 1 + cpi_diff1 + unemp_diff1 + avg_precip_diff1 + temp_diff1 + cdd_diff1))

report(auto_fit_constant)
```

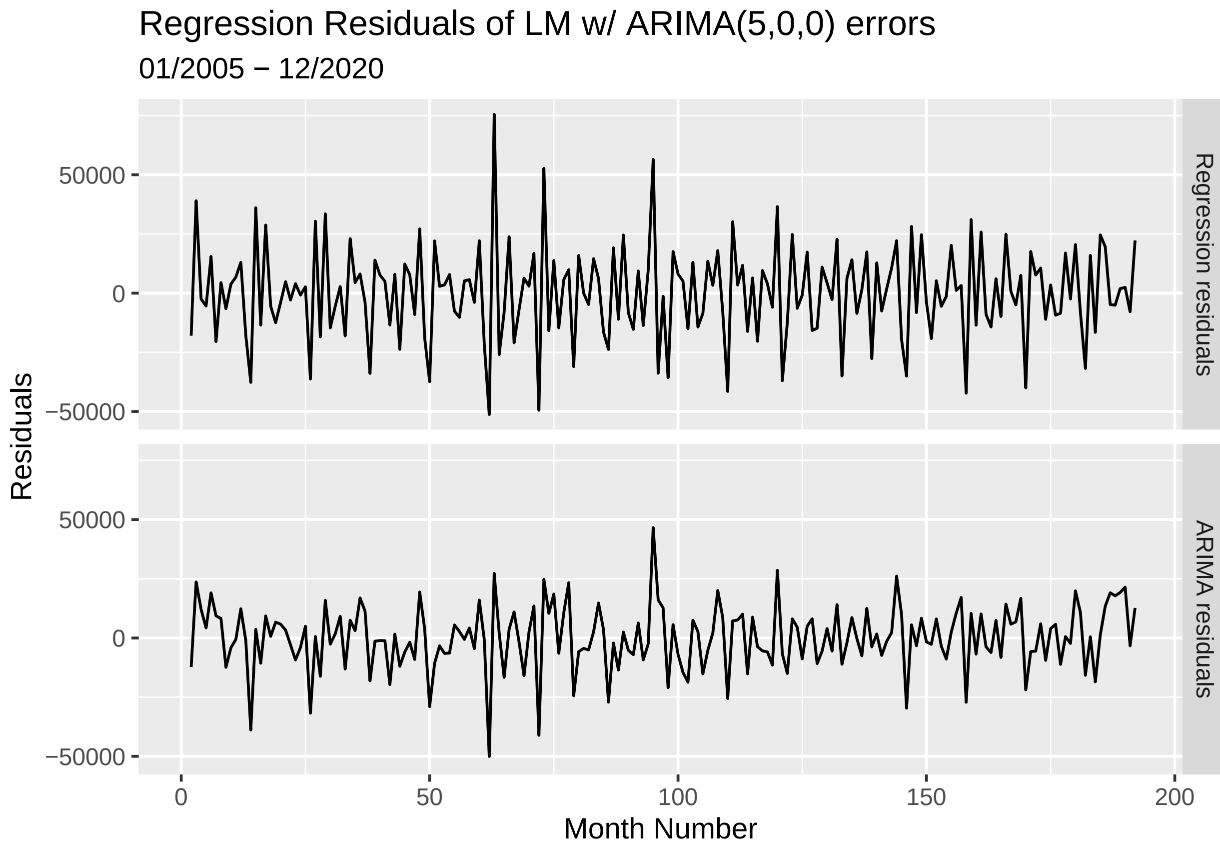
```
## Series: tw_diff1
## Model: LM w/ ARIMA(5,0,0) errors
##
```

```
## Coefficients:
##           ar1      ar2      ar3      ar4      ar5  cpi_diff1  unemp_diff1
##       -0.9113  -0.7827  -0.7225  -0.7462  -0.4107  -208.3546     89999.11
## s.e.   0.0667   0.0781   0.0811   0.0796   0.0680   211.7207     49720.01
##       avg_precip_diff1  temp_diff1  cdd_diff1  intercept
##               27786.36    956.9695  -913.0696   230.8018
## s.e.          10821.81    153.7489   536.6109   324.1543
##
## sigma^2 estimated as 189522306:  log likelihood=-2087.28
## AIC=4198.56   AICc=4200.31   BIC=4237.65
```

Regression Residuals of LM w/ ARIMA(5,0,0) errors

```
bind_rows(
    `Regression residuals` =
        as_tibble(residuals(auto_fit_constant, type = "regression")),
    `ARIMA residuals` =
        as_tibble(residuals(auto_fit_constant, type = "innovation")),
    .id = "type"
  ) %>%
  mutate(
    type = factor(type, levels=c(
      "Regression residuals", "ARIMA residuals"))
  ) %>%
  ggplot(aes(x = month_num, y = .resid)) +
  geom_line() +
  facet_grid(vars(type)) +
  labs(title = "Regression Residuals of LM w/ ARIMA(5,0,0) errors",
       subtitle = "01/2005 - 12/2020",
       x = "Month Number",
       y = "Residuals")
```

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



KPSS Test for 'total_waste'
H_0 : The time series is trend stationary vs H_a : The time series is not trend stationary

If the p-value of the test is less than some significance level (e.g. $\alpha = .05$) then we reject the null hypothesis and conclude that the time series is not trend stationary.

```r
#total waste values
nyc_ts_2 %>% features(total_waste_total, unitroot_kpss)
```

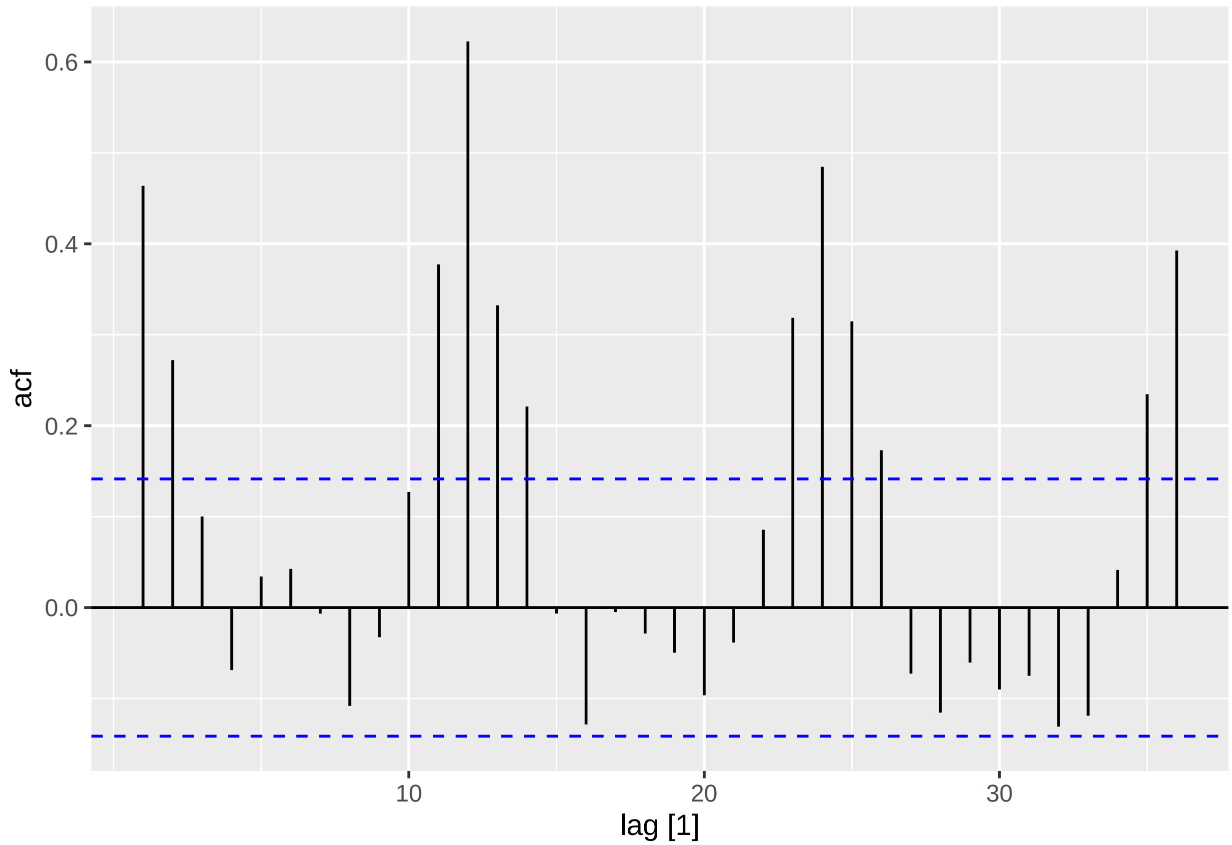
```
## # A tibble: 1 x 2
##   kpss_stat kpss_pvalue
##       <dbl>       <dbl>
## 1     0.684      0.0150
```

```r
#differenced values
nyc_ts_2 %>% features(tw_diff1, unitroot_kpss)
```

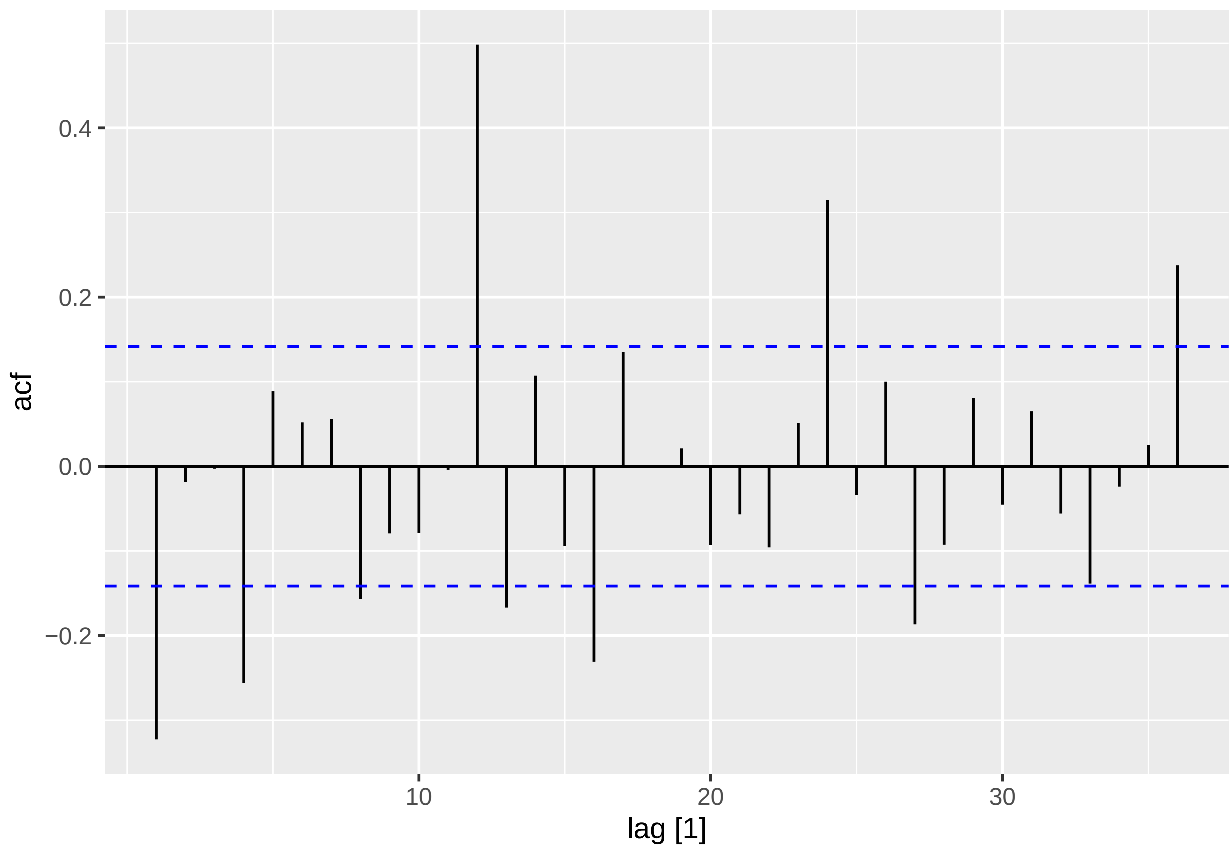
```
## # A tibble: 1 x 2
##   kpss_stat kpss_pvalue
##       <dbl>       <dbl>
## 1    0.0215         0.1
```

According to the results of the KPSS test, we reject the H_0 when evaluating the total_waste values. We fail to reject the H_0 when evaluating the differenced values

```r
nyc_ts_2 %>%
  ACF(total_waste_total, lag_max = 36) %>%
  autoplot()
```




```r
#acf of the differenced values
nyc_ts_2 %>%
  ACF(tw_diff1, lag_max = 36) %>%
  autoplot()
```



Creating ARIMA models with zoo() and the arima package from stats()

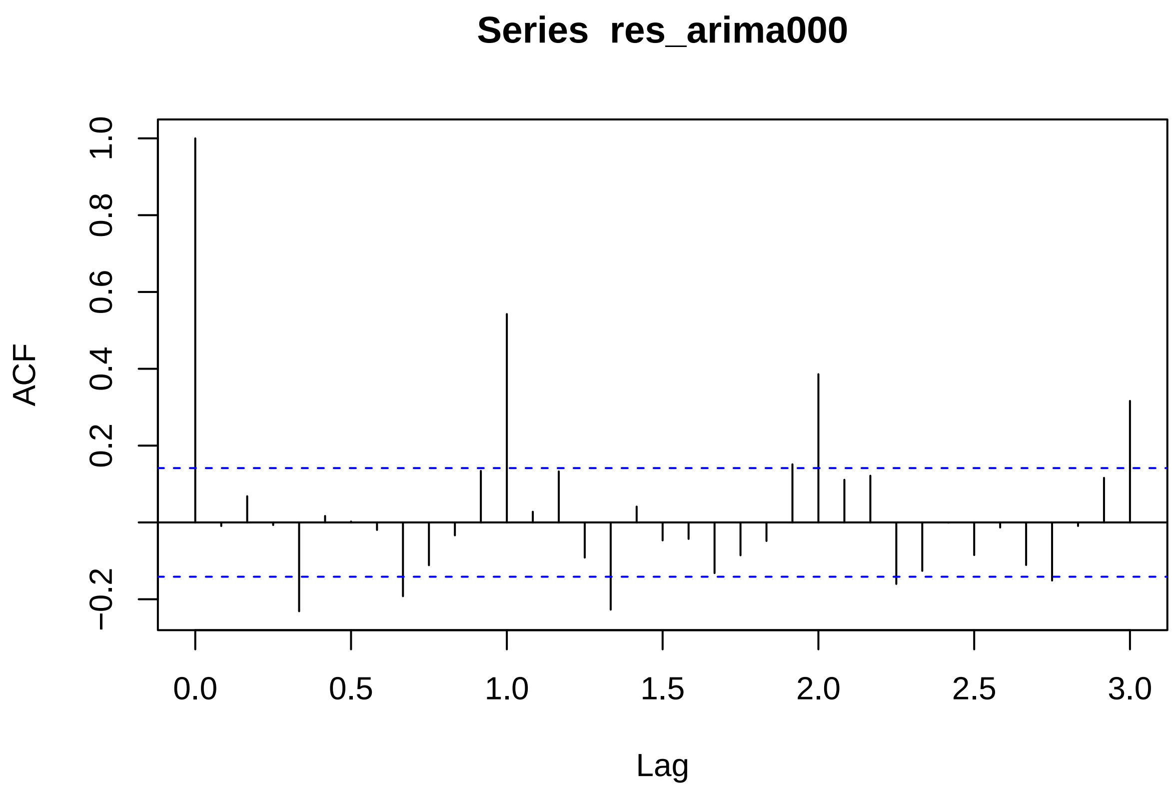
```
DSNY_NYC_zoo_ts <- ts(final_nyc_data_small[,2],
   start = as.yearmon(final_nyc_data_small$month)[1],
   frequency = 12)
# autoplot(as.zoo(DSNY_NYC_zoo_ts))
```

ARIMA(0,0,0) with constant

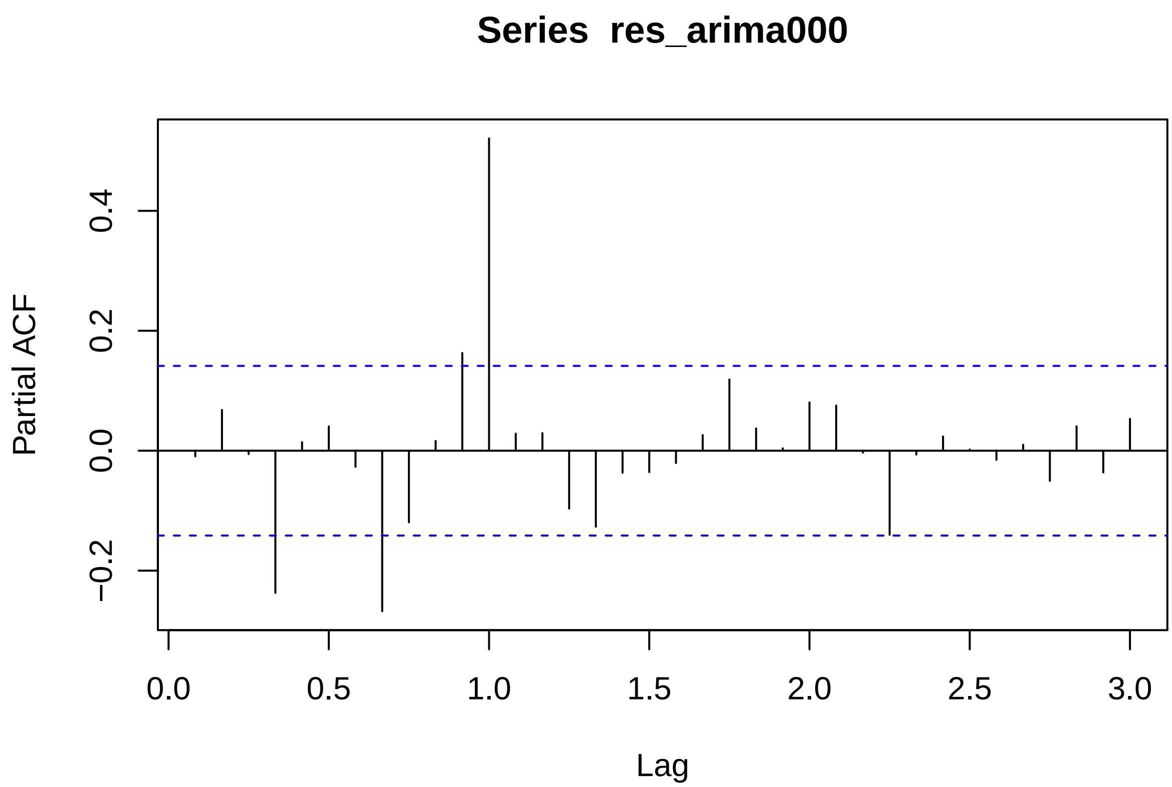
$ARIMA(0,0,0)$

```
nyc_arima000_fit_cons <- nyc_ts_2 %>%
  model(arima000_constant = ARIMA(total_waste_total ~ 1 + pdq(0,0,0)))

zoo_arima000_fit <- arima(DSNY_NYC_zoo_ts, order = 1 + c(0,0,0))
res_arima000 <- zoo_arima000_fit$residuals
acf(res_arima000, lag.max = 36)
```




```
pacf(res_arima000, lag.max = 36)
```




```
accuracy(nyc_arima000_fit_cons)[4]
```

```
## # A tibble: 1 x 1
##      RMSE
##     <dbl>
## 1 20560.
```

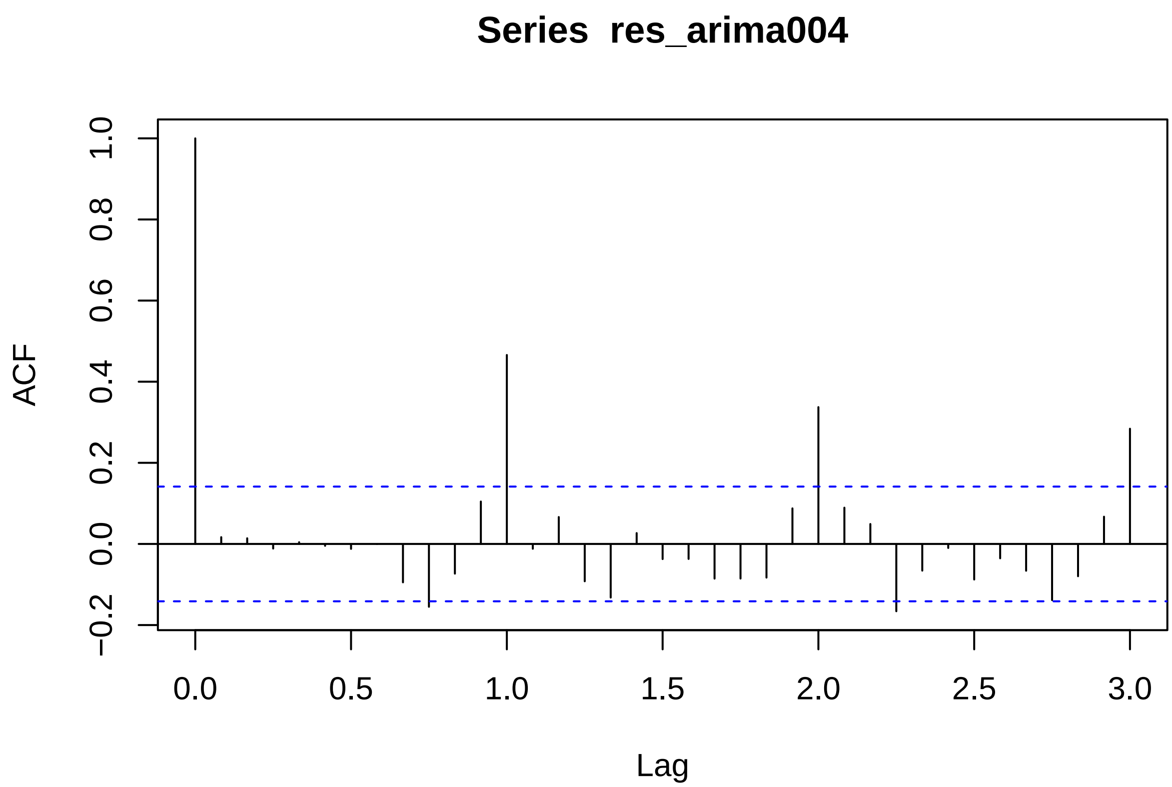
RMSE = 20560.3, The first significant lag in the ACF plot is lag 4. The first significant lag in the PACF plot is also lag 4. Both these lags are negative, which indicate the use of an MA() argument. The seasonal lags are once again present in both plots. In the PACF() plot, lag 8 is negative and significant.

ARIMA(0,0,4) with constant

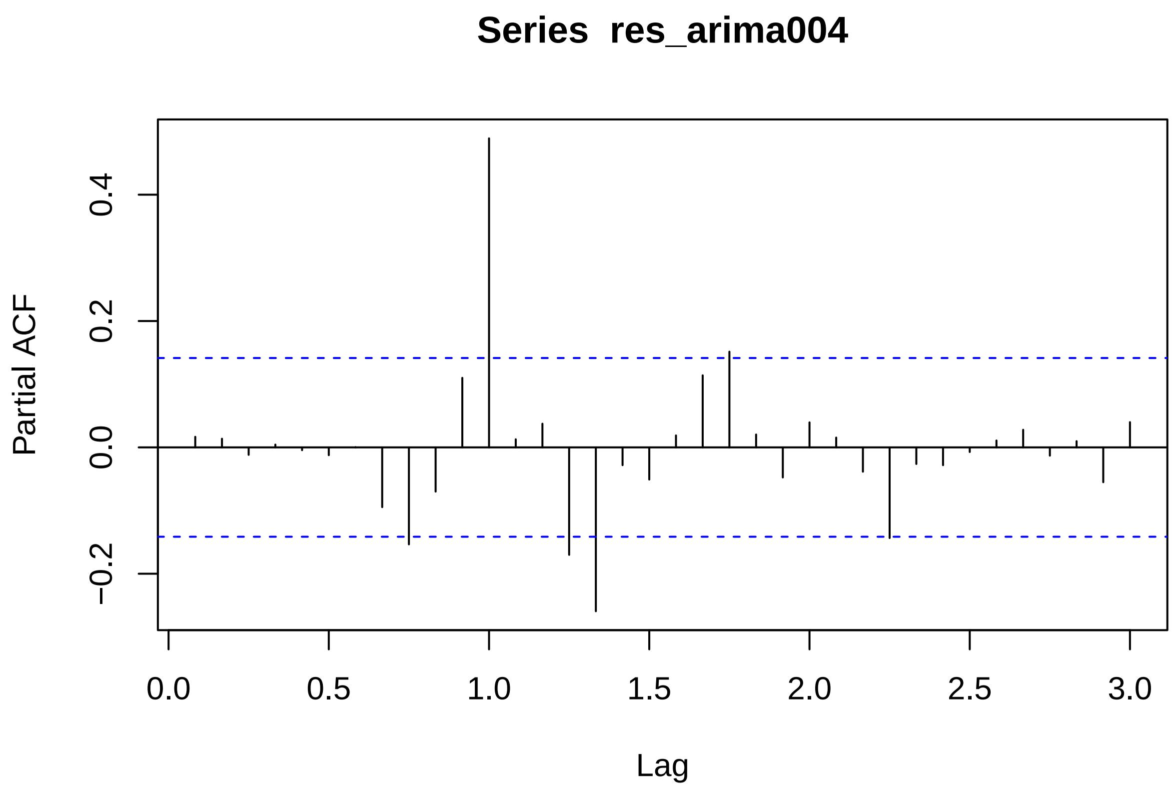
$ARIMA(0,0,4)$

```
nyc_arima004_fit_cons <- nyc_ts_2 %>%
  model(arima004_constant = ARIMA(total_waste_total ~ 1 +
                                    pdq(0,0,4)))

zoo_arima004_fit <- arima(DSNY_NYC_zoo_ts,
                          order = 1 + c(0,0,4))
#names(zoo_arima000_fit)
res_arima004 <- zoo_arima004_fit$residuals
acf(res_arima004, lag.max = 36)
```




```
pacf(res_arima004, lag.max = 36)
```




```
accuracy(nyc_arima004_fit_cons)[4]
```

```
## # A tibble: 1 x 1
##     RMSE
##    <dbl>
## 1 17603.
```

RMSE = 17602.62. Seasonal lags worth exploring. In the PACF plot, lag = 9 is negatively autocorrelated and significant

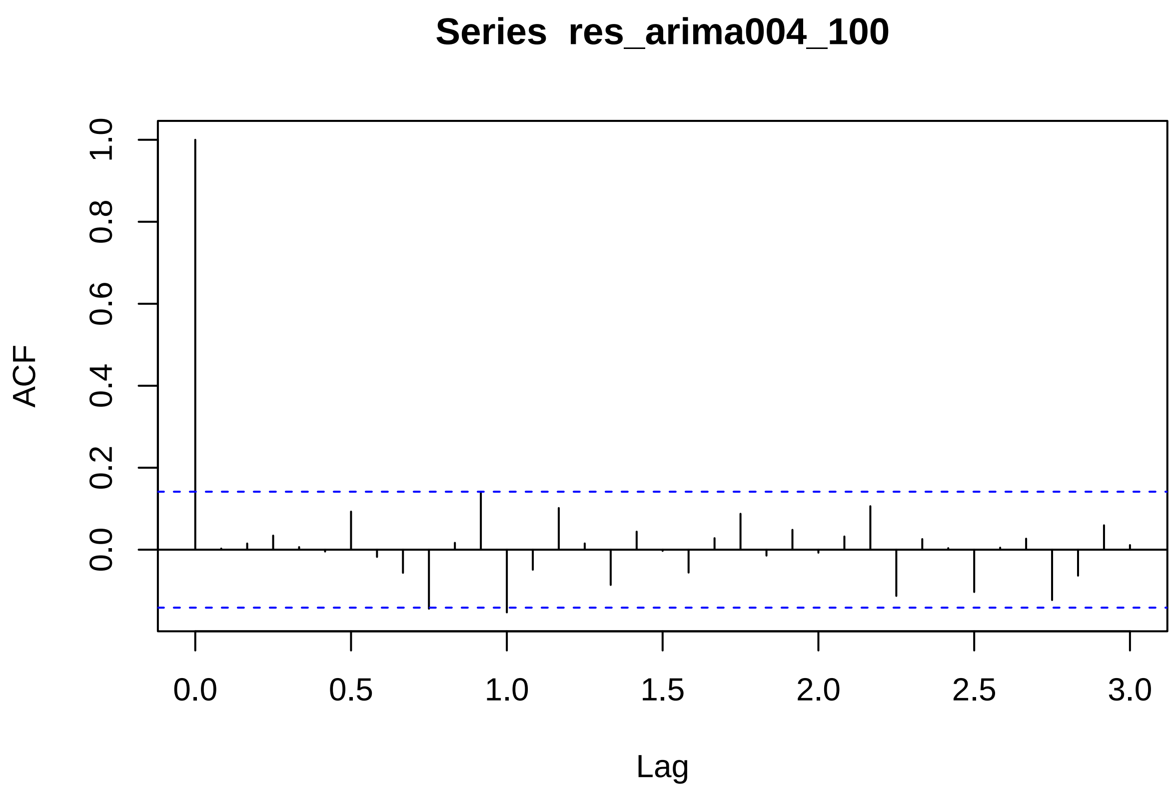
ARIMA(0,0,4)(1,0,0) with constant and seasonal

$ARIMA(0,0,4)(1,0,0)_{12}$

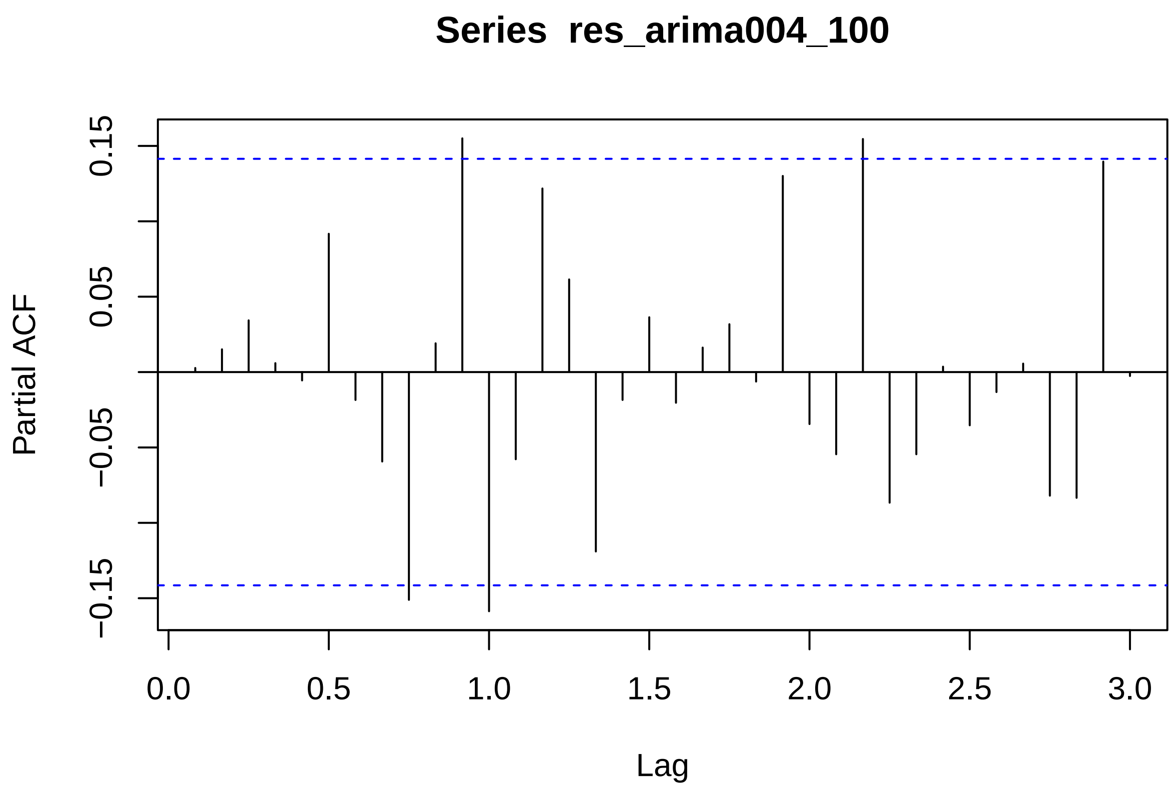
```
nyc_arima004_100_seasonal_fit_cons <- nyc_ts_2 %>%
  model(arima004_100_constant = ARIMA(total_waste_total ~ 1 +
                                  pdq(0,0,4) +
                                  PDQ(1,0,0,
                                      period = 12)))

zoo_arima004_100_seasonalfit <- arima(DSNY_NYC_zoo_ts,
                          order = 1 + c(0,0,4),
                          seasonal = list(order = c(1, 0L, 0L),
                                          period = 12))
```

```r
#names(zoo_arima000_fit)
res_arima004_100 <- zoo_arima004_100_seasonalfit$residuals
acf(res_arima004_100, lag.max = 36)
```




```r
pacf(res_arima004_100, lag.max = 36)
```




```
accuracy(nyc_arima004_100_seasonal_fit_cons)[4]
```

```
## # A tibble: 1 x 1
##      RMSE
##     <dbl>
## 1 13927.
```

RMSE = 13926.64. Most of the lags are now bounded between -0.15 and 0.15. In the PACF, the first significant lag is lag = 9, which is barely significant and negative. Lag = 12 is also significant.

we can also work with the differenced values, we will create some models with them

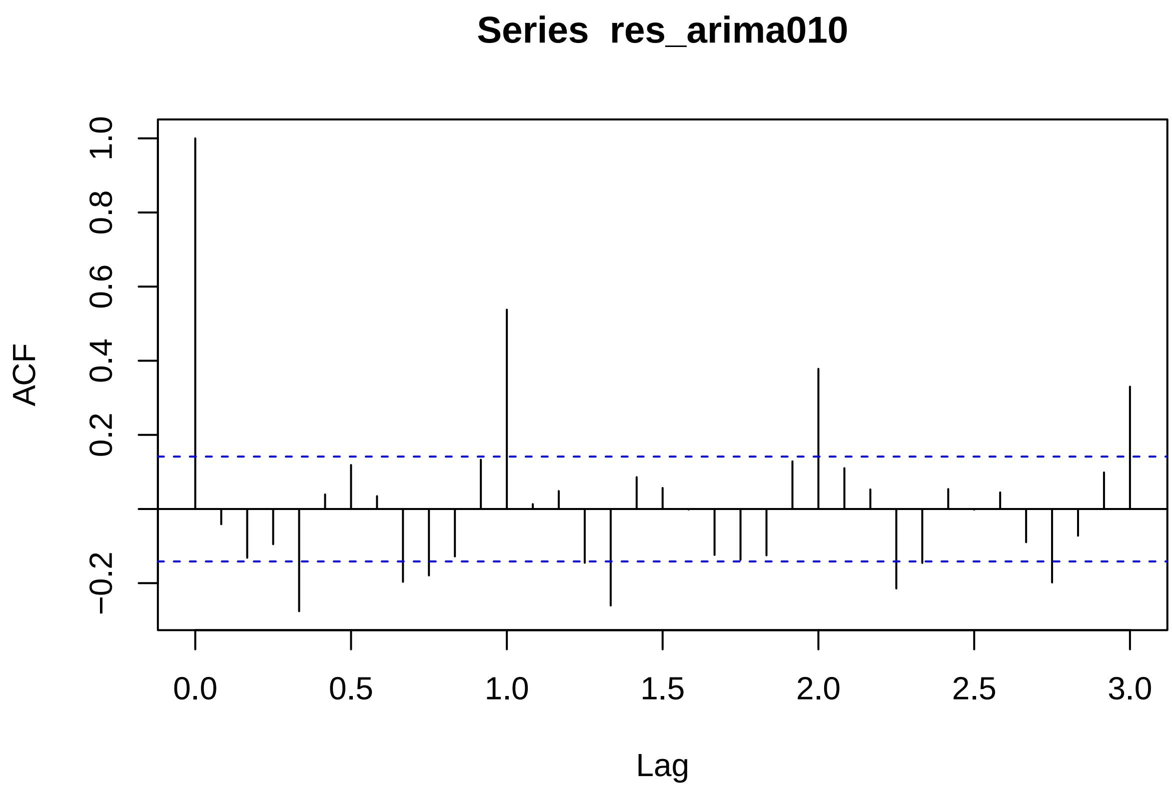
ARIMA(0,1,0) with constant

$ARIMA(0,1,0)$

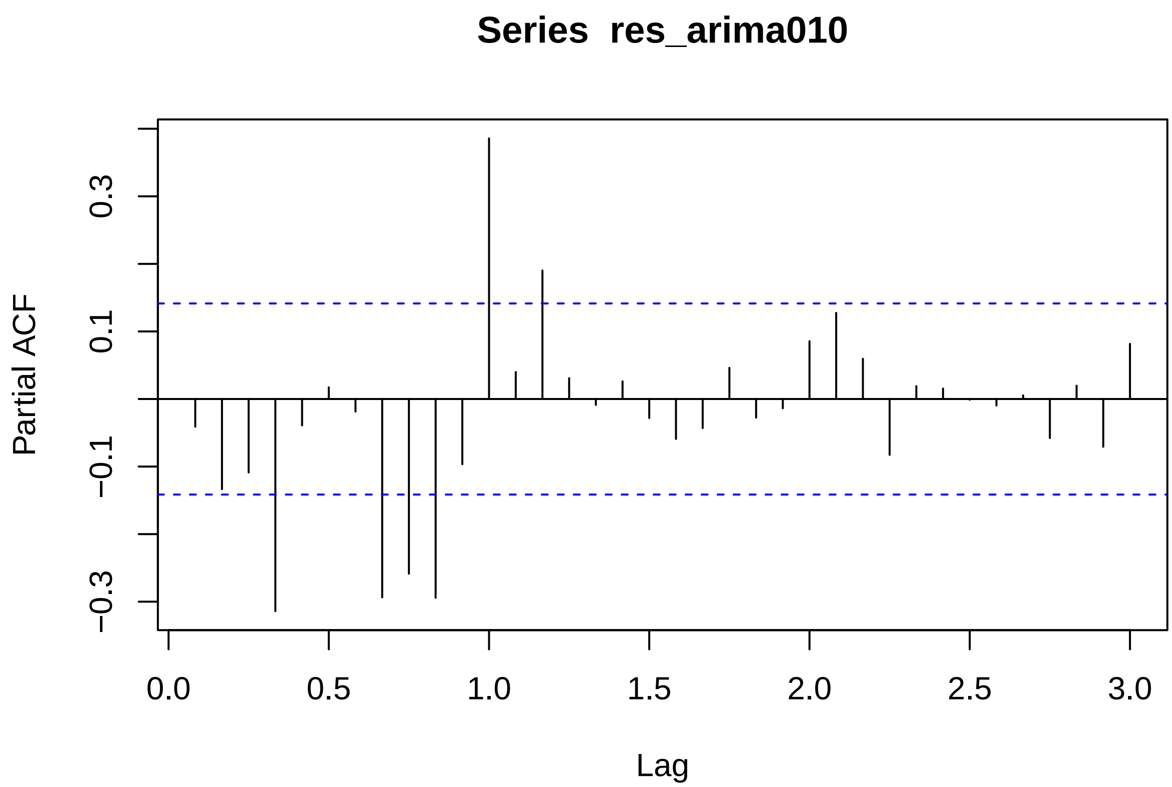
```
nyc_arima010_fit_cons <- nyc_ts_2 %>%
  model(arima010_constant = ARIMA(total_waste_total ~ 1 + pdq(0,1,0)))

zoo_arima010_fit <- arima(DSNY_NYC_zoo_ts,
                          order = 1 + c(0,1,0))

res_arima010 <- zoo_arima010_fit$residuals
acf(res_arima010, lag.max = 36)
```




```
pacf(res_arima010, lag.max = 36)
```




```
accuracy(nyc_arima010_fit_cons)[4]
```

```
## # A tibble: 1 x 1
##     RMSE
##    <dbl>
## 1 21209.
```

RMSE = 21209.02. In the ACF plot, the seasonal lags are once again significant. Looking at the PACF, the first significant lag is 4, and is negatively autocorrelated.

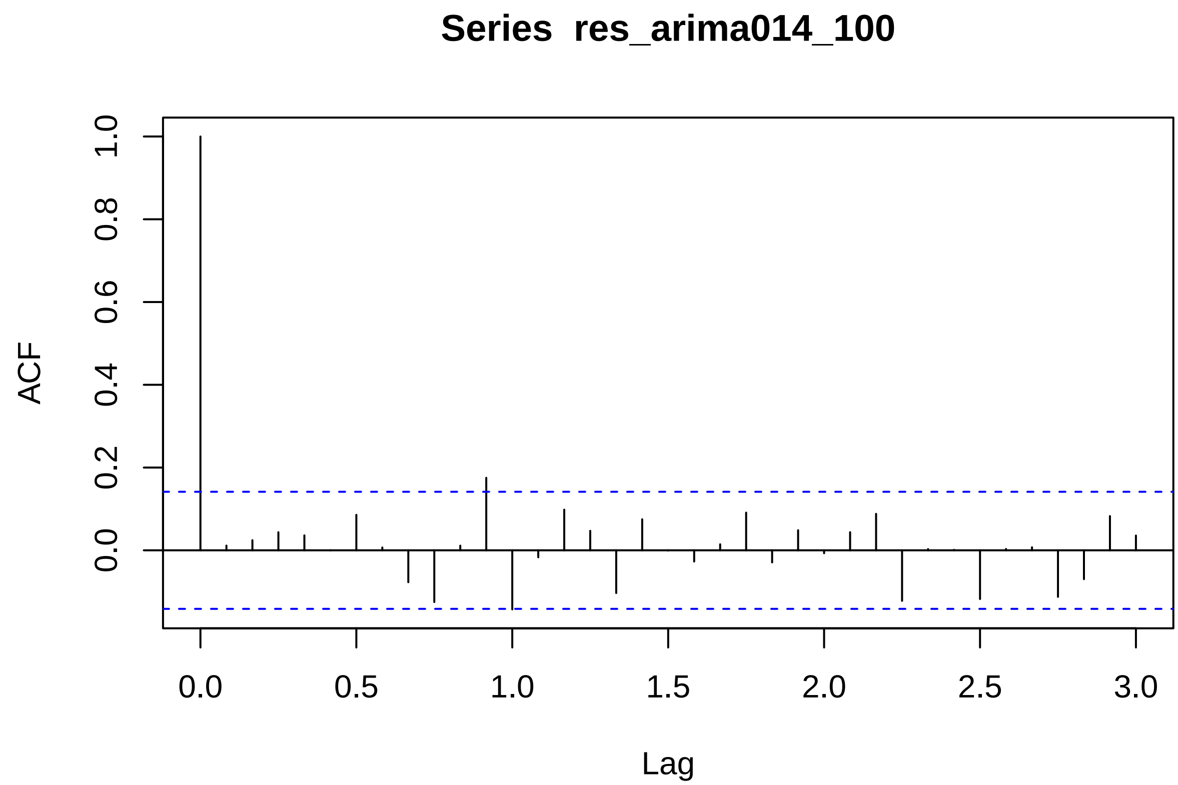
ARIMA(0,1,4)(1,0,0) with constant and seasonal

$ARIMA(0,1,4)(1,0,0)[12]$

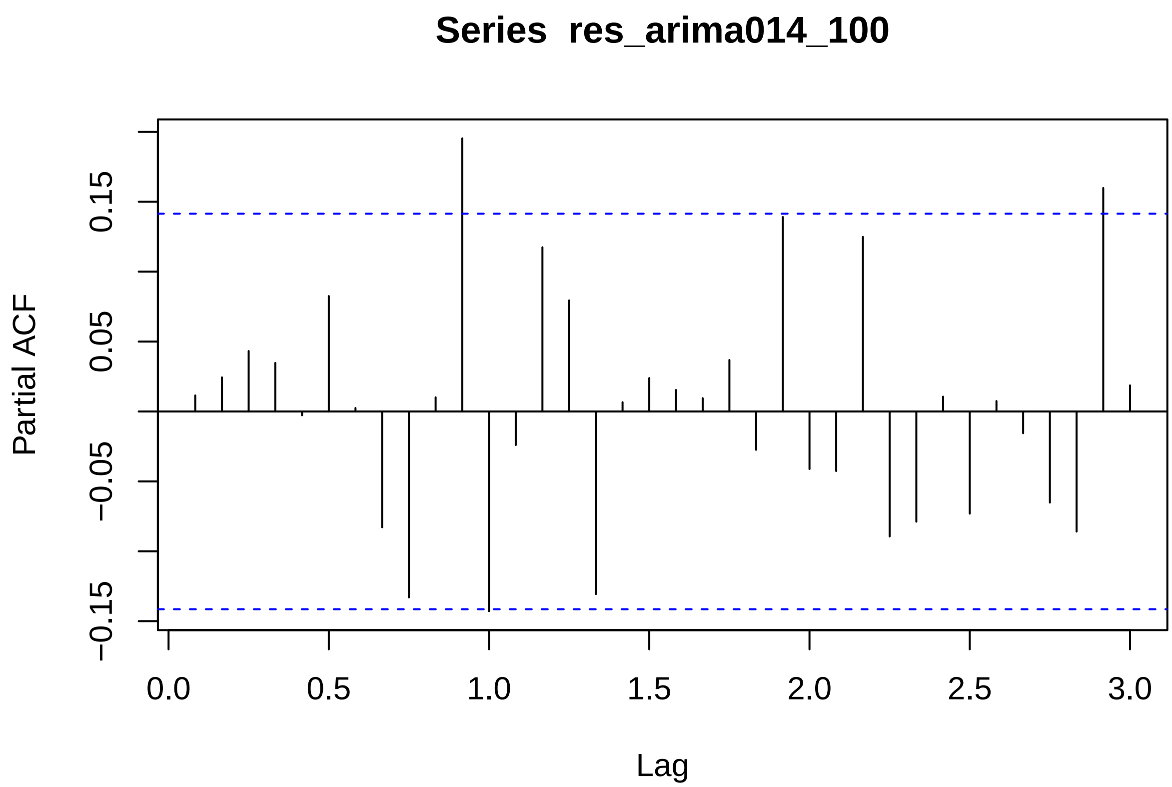
```
nyc_arima014_100_fit_cons <- nyc_ts_2 %>%
  model(arima014_100_constant = ARIMA(total_waste_total ~ 1 +
                                      pdq(0,1,4) +
                                      PDQ(1,0,0,
                                          period = 12)))

zoo_arima014_100_fit <- arima(DSNY_NYC_zoo_ts,
                              order = 1 + c(0,1,4),
                              seasonal = list(order = c(1,0L,0L),
                                              period = 12))
```

```r
res_arima014_100 <- zoo_arima014_100_fit$residuals
acf(res_arima014_100, lag.max = 36)
```




```r
pacf(res_arima014_100, lag.max = 36)
```




```r
accuracy(nyc_arima014_100_fit_cons)[4]
```

```
## # A tibble: 1 x 1
##     RMSE
##    <dbl>
## 1 13817.
```

RMSE = 13816.72. In both plots, lag = 11. Is the lag that is the most autocorrelated. The RMSE has decreasaed when compared to $ARIMA(0,0,4)(1,0,0)_{12}$

Auto-arima

For our final model, we will look and compare the results of an auto-arima model from the feasts package.

```r
nyc_auto_arima_fit_cons <- nyc_ts_2 %>%
  model(stepwise = ARIMA(total_waste_total),
        search = ARIMA(total_waste_total,
                       stepwise = FALSE,
                       approximation = FALSE))

accuracy(nyc_auto_arima_fit_cons)[1:4]
```

```
## # A tibble: 2 x 4
```

```
##    .model   .type       ME   RMSE
##    <chr>    <chr>    <dbl>  <dbl>
## 1 stepwise Training -164.  16944.
## 2 search   Training  -85.1 16990.
```

The stepwise model has RMSE = 16493.55, while the search model has RMSE = 16990.13. We will take a look at the ACF and PACF plots of the stepwise model.

```
nyc_auto_arima_fit_cons %>% select(.model = stepwise) %>% report()
```

```
## Series: total_waste_total
## Model: ARIMA(2,1,4)
##
## Coefficients:
##          ar1      ar2      ma1     ma2      ma3      ma4
##       0.7199  -0.7437  -1.3232  1.1658  -0.6453  -0.1056
## s.e.  0.1220   0.0781   0.1490  0.1771   0.1428   0.1056
##
## sigma^2 estimated as 297946445:  log likelihood=-2132.68
## AIC=4279.35   AICc=4279.96   BIC=4302.12
```

```
# print("--------------")
# nyc_auto_arima_fit_cons %>% select(.model = search) %>% report()
```

The AICc, AIC, BIC metrics for the stepwise model is barely greater than the metrics for the search model.

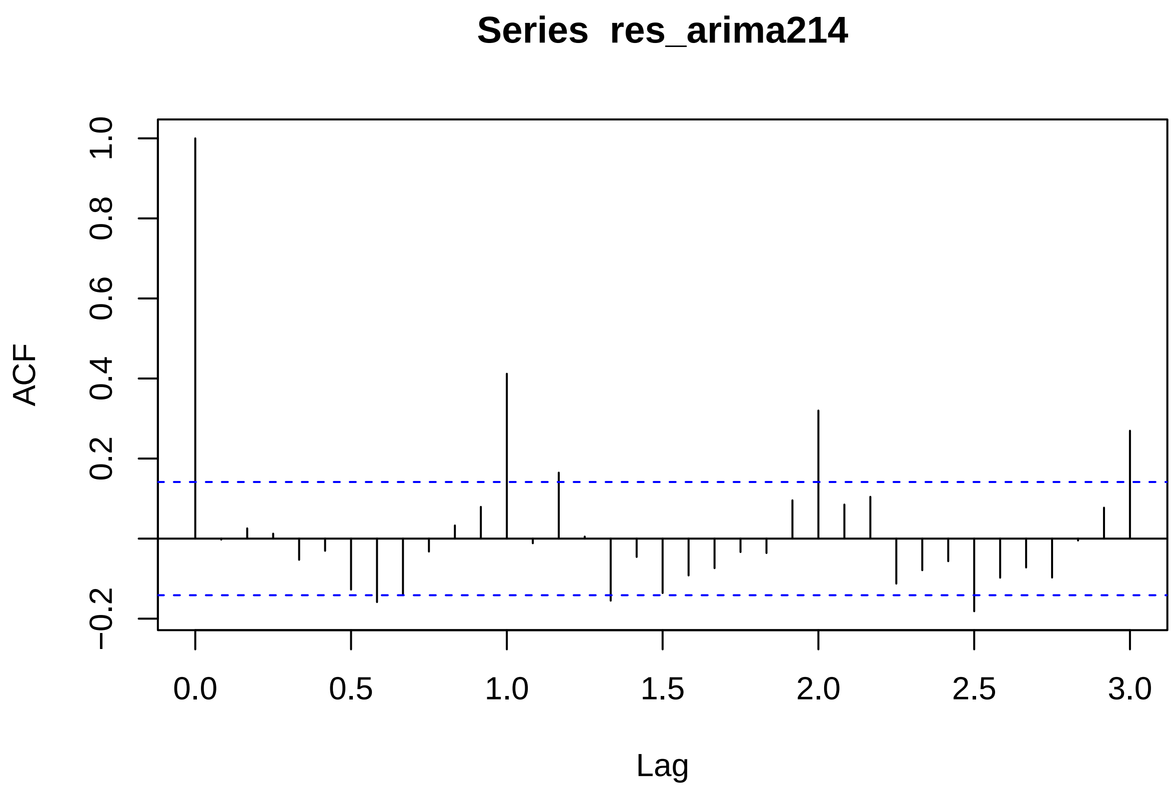
ARIMA(2,1,4) with constant from auto-arima

$ARIMA(2,1,4)$

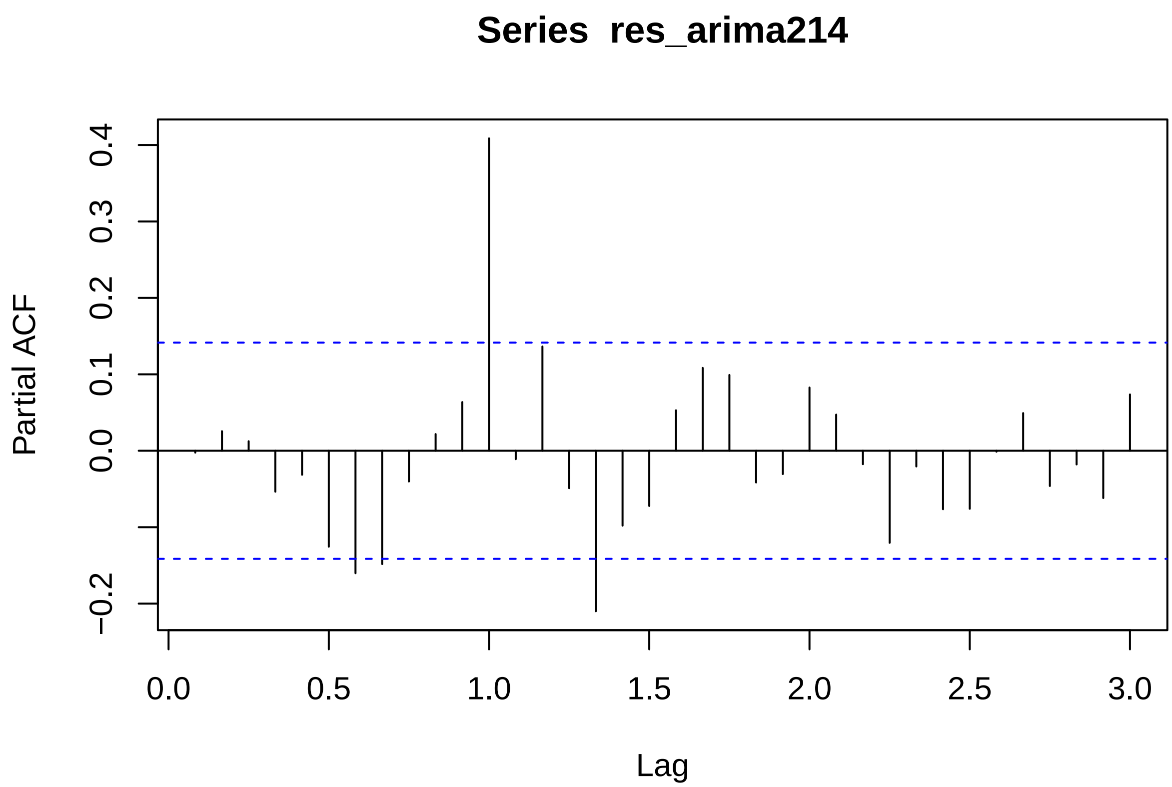
```
nyc_arima214_fit_cons <- nyc_ts_2 %>%
  model(arima214_constant = ARIMA(total_waste_total ~ 1 +
                                      pdq(2,1,4)))

zoo_arima214_fit <- arima(DSNY_NYC_zoo_ts,
                              order = 1 + c(2,1,4))

res_arima214 <- zoo_arima214_fit$residuals
acf(res_arima214, lag.max = 36)
```




```
pacf(res_arima214, lag.max = 36)
```




```
accuracy(nyc_arima214_fit_cons)[4]
```

```
## # A tibble: 1 x 1
##      RMSE
##     <dbl>
## 1 16937.
```

Summary of total waste ARIMA Models

$ARIMA(0,0,4)(1,0,0)[12]$ has RMSE = 13926
$ARIMA(0,1,4)(1,0,0)[12]$ has RMSE = 13816 $ARIMA(2,1,4)$ is the step model and has RMSE = 16936.99

Dynamic Regression models with ARIMA models from before

$\#\# \, textARIMA(0,0,4)(1,0,0)[12]$

```r
dr_diff_cons_fit1 <- nyc_ts_2 %>%
  model(dynam_regress_diff1 = ARIMA(tw_diff1 ~ 1 + cpi_diff1 + unemp_diff1 +
                avg_precip_diff1 + temp_diff1 +
                cdd_diff1 +
                trend() +
                pdq(0,0,4) +
                PDQ(1,0,0,
                    period = 12)))

report(dr_diff_cons_fit1)
```

```
## Series: tw_diff1
## Model: LM w/ ARIMA(0,0,4)(1,0,0)[12] errors
##
## Coefficients:
##           ma1      ma2     ma3      ma4    sar1  cpi_diff1  unemp_diff1
##       -1.1023  -0.0424  0.1890  -0.0444  0.5558   -12.2330    115886.55
## s.e.   0.0752   0.1125  0.1141   0.0713  0.0670   111.3998     25901.03
##       avg_precip_diff1  temp_diff1   cdd_diff1  trend()  intercept
##              38605.720   1019.2643  -1117.2799   5.9850  -651.2312
## s.e.          8853.597    155.9817    499.2012   0.8808   178.8632
##
## sigma^2 estimated as 136438316:  log likelihood=-2058.56
## AIC=4143.12   AICc=4145.17   BIC=4185.47
```

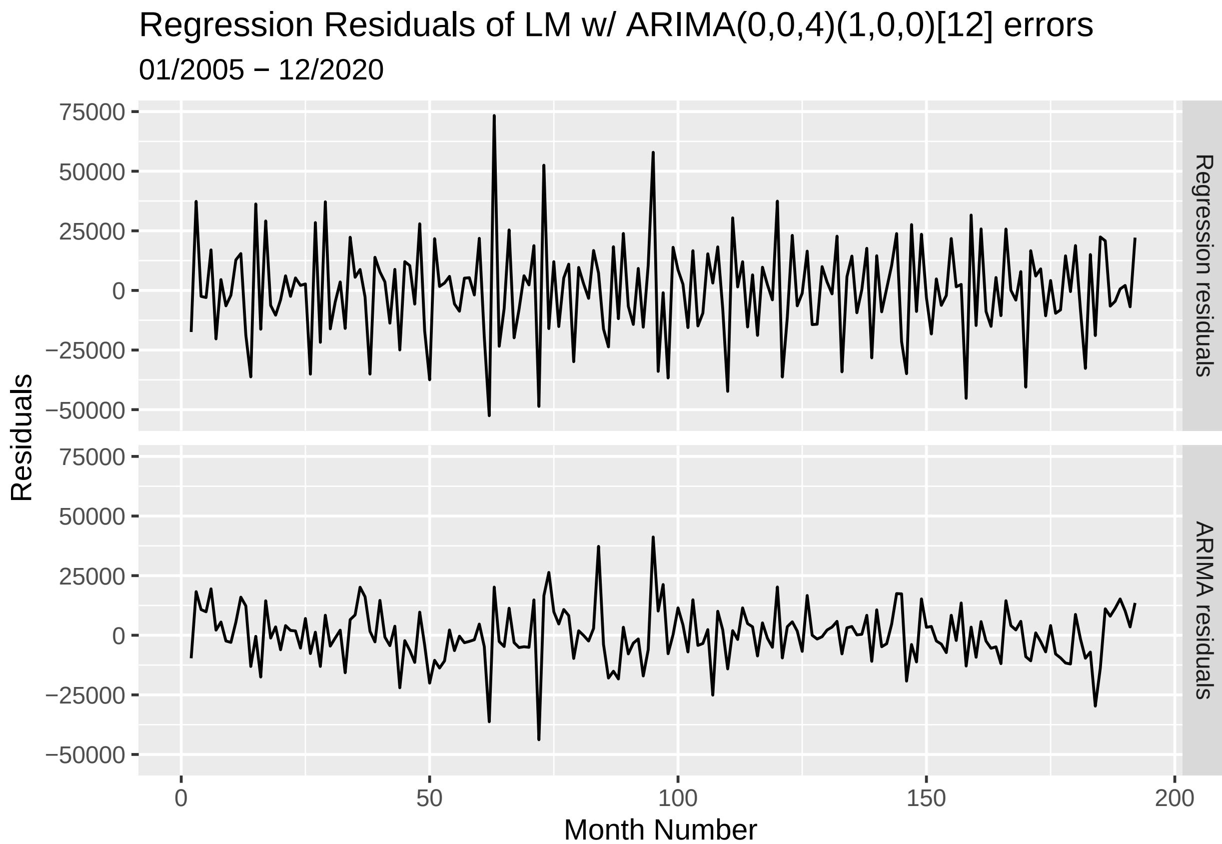
```r
accuracy(dr_diff_cons_fit1)
```

```
## # 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 dynam_regress_diff1 Traini~  247. 11339. 8529.  8.96  246. 0.322 0.327 0.00194
```

When adding the trend() argument, the coefficients of the majority of the predictors and arima errors decrease.

```r
  bind_rows(
    `Regression residuals` =
        as_tibble(residuals(dr_diff_cons_fit1, type = "regression")),
    `ARIMA residuals` =
        as_tibble(residuals(dr_diff_cons_fit1, type = "innovation")),
    .id = "type"
  ) %>%
  mutate(
    type = factor(type, levels=c(
      "Regression residuals", "ARIMA residuals"))
  ) %>%
  ggplot(aes(x = month_num, y = .resid)) +
  geom_line() +
  facet_grid(vars(type)) +
  labs(title = "Regression Residuals of LM w/ ARIMA(0,0,4)(1,0,0)[12] errors",
      subtitle = "01/2005 - 12/2020",
      x = "Month Number",
      y = "Residuals")
```

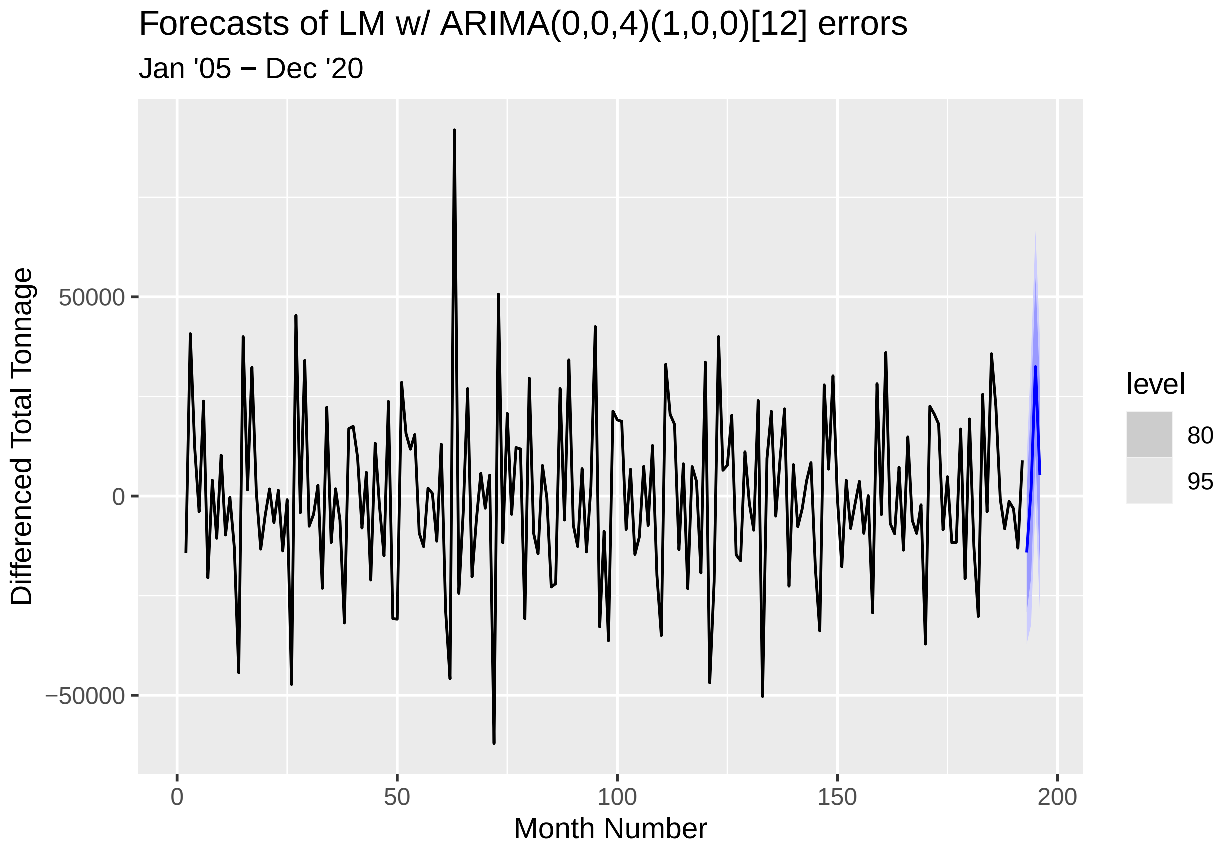
```
## Warning: Removed 1 row(s) containing missing values (geom_path).
```




```
nyc_data_small_future_diff <- new_data(nyc_ts_2,4) %>%
  mutate(cpi_diff1 = c(825.413,111, 831.067, 836.885),
         unemp_diff1 = c(0.1333,0.1280, 0.1130, 0.1090),
         avg_precip_diff1 = c(0.07, 0.18, 0.17, 0.12),
         temp_diff1 = c(-4.4, -0.6, 11.6, 8.8),
         cdd_diff1 = c(0,0,0.1, 0.1))
```

```
forecast(dr_diff_cons_fit1, new_data = nyc_data_small_future_diff) %>%
  autoplot(nyc_ts_1) +
  labs(x  = "Month Number",
       y = "Differenced Total Tonnage",
       title = "Forecasts of LM w/ ARIMA(0,0,4)(1,0,0)[12] errors",
       subtitle = "Jan '05 - Dec '20")
```

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



$ARIMA(0, 1, 4)(1, 0, 0)[12]$

```
dr_diff_cons_fit2 <- nyc_ts_2 %>%
  model(dynam_regress_diff2 = ARIMA(tw_diff1 ~ 1 + cpi_diff1 + unemp_diff1 +
              avg_precip_diff1 + temp_diff1 +
              cdd_diff1 +
              pdq(0,1,4) +
              PDQ(1,0,0,
                  period = 12)))
```

```
## Warning: 1 error encountered for dynam_regress_diff2
## [1] system is computationally singular: reciprocal condition number = 2.10254e-16
```

```
report(dr_diff_cons_fit2)
```

```
## Series: tw_diff1
## Model: NULL model
## NULL model
```

```
accuracy(dr_diff_cons_fit2)
```

```
## # 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 dynam_regress_diff2 Training   NaN   NaN   NaN   NaN   NaN   NaN   NaN    NA
```

Was given an error: Warning: Provided exogenous regressors are rank deficient, removing regressors: `trend()` Warning: 1 error encountered for dynam_regress_diff2 [1] system is computationally singular: reciprocal condition number = 2.10254e-16

https://stats.stackexchange.com/questions/76488/error-system-is-computationally-singular-when-running-a-glm

ARIMA(2,1,4)

```
dr_diff_cons_fit3 <- nyc_ts_2 %>%
  model(dynam_regress_diff3 = ARIMA(tw_diff1 ~ 1 + cpi_diff1 + unemp_diff1 +
                avg_precip_diff1 + temp_diff1 +
                cdd_diff1 +
                pdq(2,1,4)))
```

```
## Warning in sqrt(diag(best$var.coef)): NaNs produced
```

```
report(dr_diff_cons_fit3)
```

```
## Series: tw_diff1
## Model: LM w/ ARIMA(2,1,4) errors
##
## Coefficients:
##           ar1      ar2      ma1      ma2     ma3     ma4  cpi_diff1
##       -0.7215  -0.0912  -1.2477  -0.4101  0.6364  0.0502    10.2995
## s.e.      NaN      NaN      NaN      NaN  0.0481     NaN   134.6692
##       unemp_diff1  avg_precip_diff1  temp_diff1   cdd_diff1  intercept
##          86272.54          34654.71   1100.6274  -1524.7383     9.2765
## s.e.     36640.66          12044.14    125.5957    403.7047    16.6183
##
## sigma^2 estimated as 208519204:  log likelihood=-2090.36
## AIC=4206.73   AICc=4208.78   BIC=4249
```

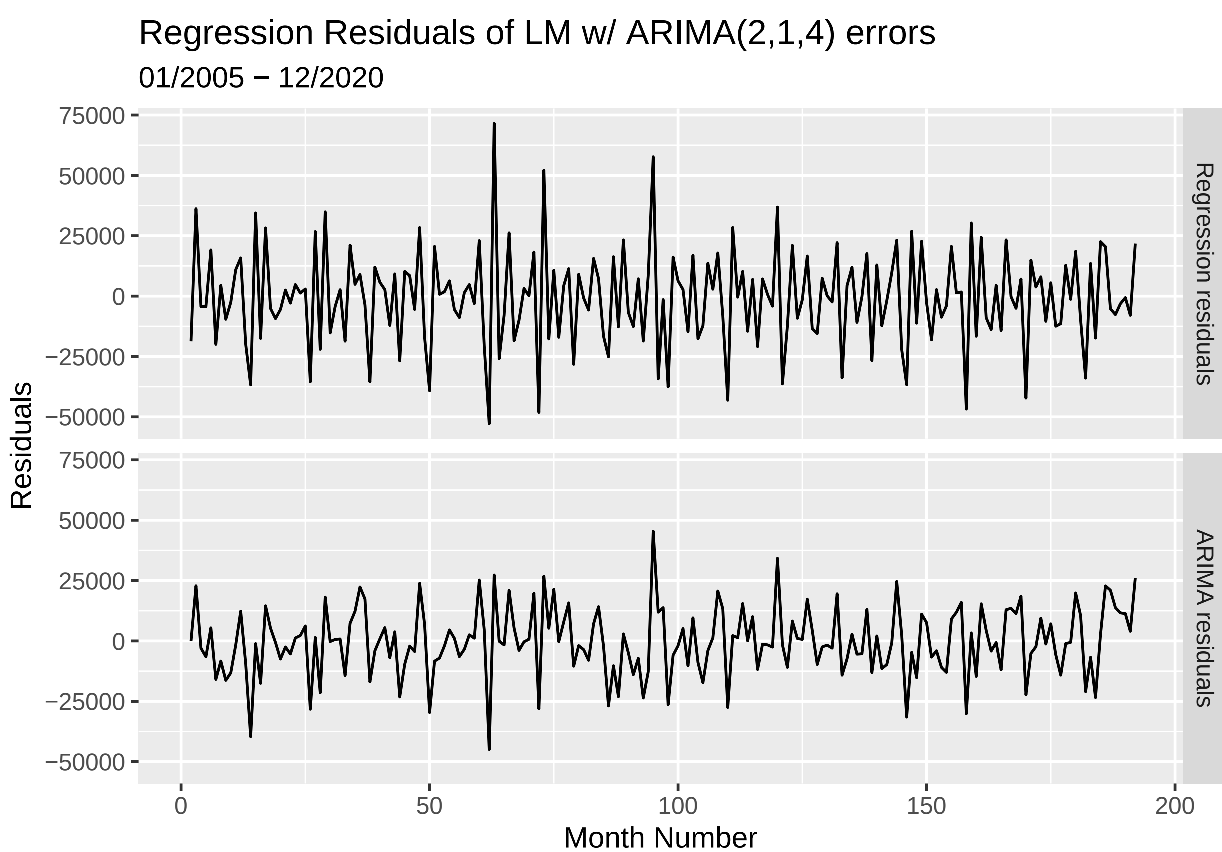
```
accuracy(dr_diff_cons_fit3)
```

```
## # 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 dynam_regress_diff3 Trai~ -271. 13979. 10551.  53.7  256. 0.399 0.403 -0.00494
```

When including the trend() parameter, I get this error: Warning: Provided exogenous regressors are rank deficient, removing regressors: `trend()` And this error is also returned: Warning in sqrt(diag(best$var.coef)) : NaNs produced

```
  bind_rows(
    `Regression residuals` =
        as_tibble(residuals(dr_diff_cons_fit3, type = "regression")),
    `ARIMA residuals` =
        as_tibble(residuals(dr_diff_cons_fit3, type = "innovation")),
    .id = "type"
  ) %>%
  mutate(
    type = factor(type, levels=c(
      "Regression residuals", "ARIMA residuals"))
  ) %>%
  ggplot(aes(x = month_num, y = .resid)) +
  geom_line() +
  facet_grid(vars(type)) +
  labs(title = "Regression Residuals of LM w/ ARIMA(2,1,4) errors",
       subtitle = "01/2005 - 12/2020",
       x = "Month Number",
       y = "Residuals")
```

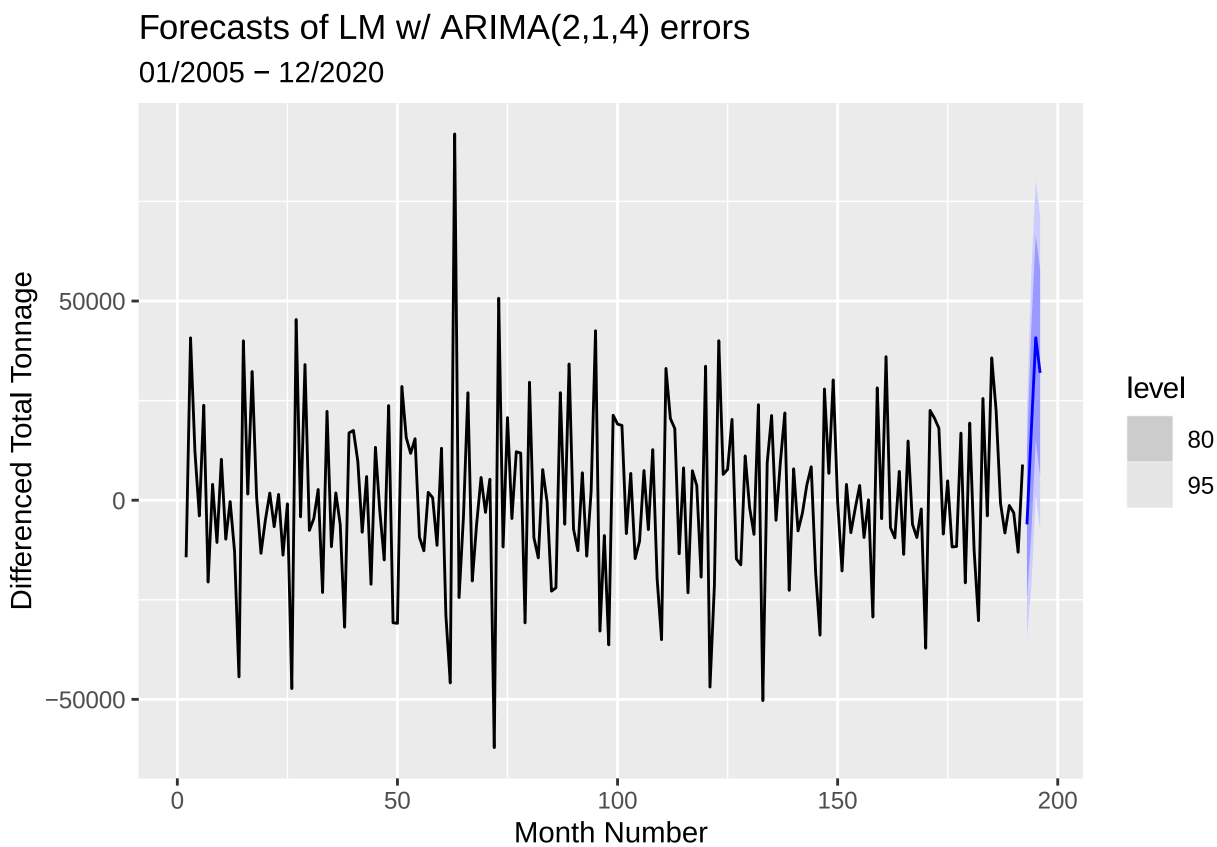
```
## Warning: Removed 1 row(s) containing missing values (geom_path).
```




```
forecast(dr_diff_cons_fit3, new_data = nyc_data_small_future_diff) %>%
  autoplot(nyc_ts_1) +
  labs(x  = "Month Number",
```

```
       y = "Differenced Total Tonnage",
       title = "Forecasts of LM w/ ARIMA(2,1,4) errors",
       subtitle = "01/2005 - 12/2020")
```

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



Summary of Dynamic Regression models

LM w/ ARIMA(5,0,0) errors has AICc=4198.5
LM w/ ARIMA(0,0,4)(1,0,0)[12] errors has RMSE = 11339.34 and AICc=4145.17
LM w/ ARIMA(0,1,4)(1,0,0)[12] does not give us a output
LM w/ ARIMA(2,1,4) errors has RMSE = 13979.22 and AICc = 4208.78