Global CO_2 Emissions

Edgar Leon, Long Vu, Cameron Yenche

Introduction

The Keeling Curve

In the 1950s, the geochemist Charles David Keeling observed a seasonal pattern in the amount of carbon dioxide present in air samples collected over the course of several years. He was able to attribute this pattern to the variation in global rates of photosynthesis throughout the year, caused by the difference in land area and vegetation cover between the Earth's northern and southern hemispheres.

In 1958 Keeling began continuous monitoring of atmospheric carbon dioxide concentrations from the Mauna Loa Observatory in Hawaii. Mauna Loa was chosen as a long-term monitoring site due to its remote location far from continents and its lack of vegetation. Keeling soon observed a trend increase in carbon dioxide (CO2) levels in addition to the seasonal cycle. He was able to attribute this trend increase to growth in global rates of fossil fuel combustion.

Since CO2 is a greenhouse gas, higher levels of CO2 can lead to a warmer planet, causing climate change. Climate change can result in more frequent and severe weather events such as floods, cyclones, typhoons, and wildfires. It can also lead to a decrease in crop yields and put many animal species at risk of extinction.

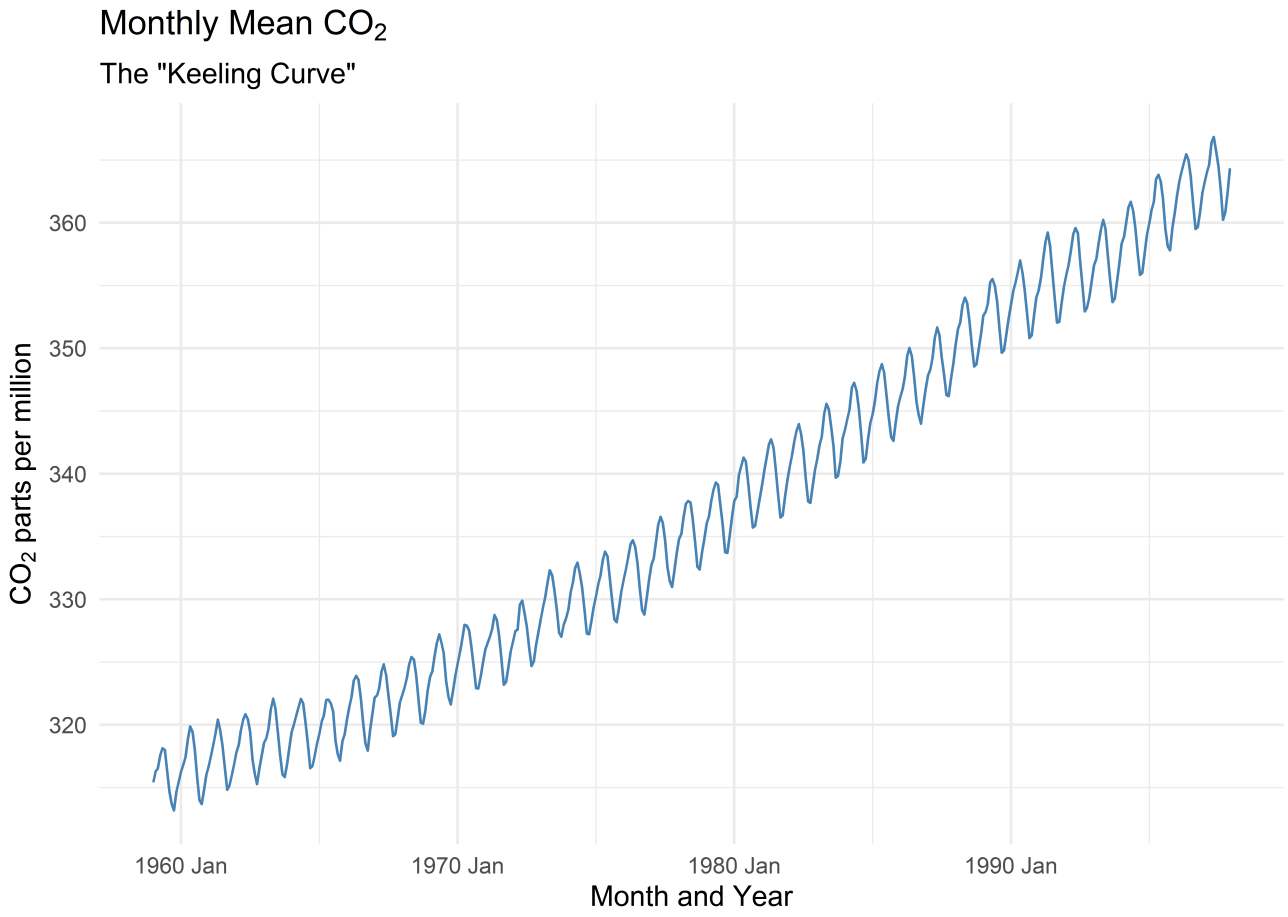
Our team will analyze the CO2 average parts per million recorded from Manua Loa since 1957 to the present (1997).

The apparent increasing trend could be dismissed as a transient stochastic phenomenon, if so, it should be possible to model the trend using stochastic functions and without the use of deterministic functions.

Our team will fit the data using stochastic ARIMA models, and deterministic Linear Time Trend models and and use diagnostics to determine which one best fits the data. Our team

will also forecast what the expected CO2 levels wil be in the 2020 and 2022, if the same trends continue to hold.

Our exercise will aid the scientific community to decide if the increasing trend can be dismissed or should further research be made. Furthermore, our forecasts will aid policy makers to take the steps to prevent the increasing trend of CO2 levels in the atmosphere.



CO2 data

Background and Exploratory Data Analysis (EDA)

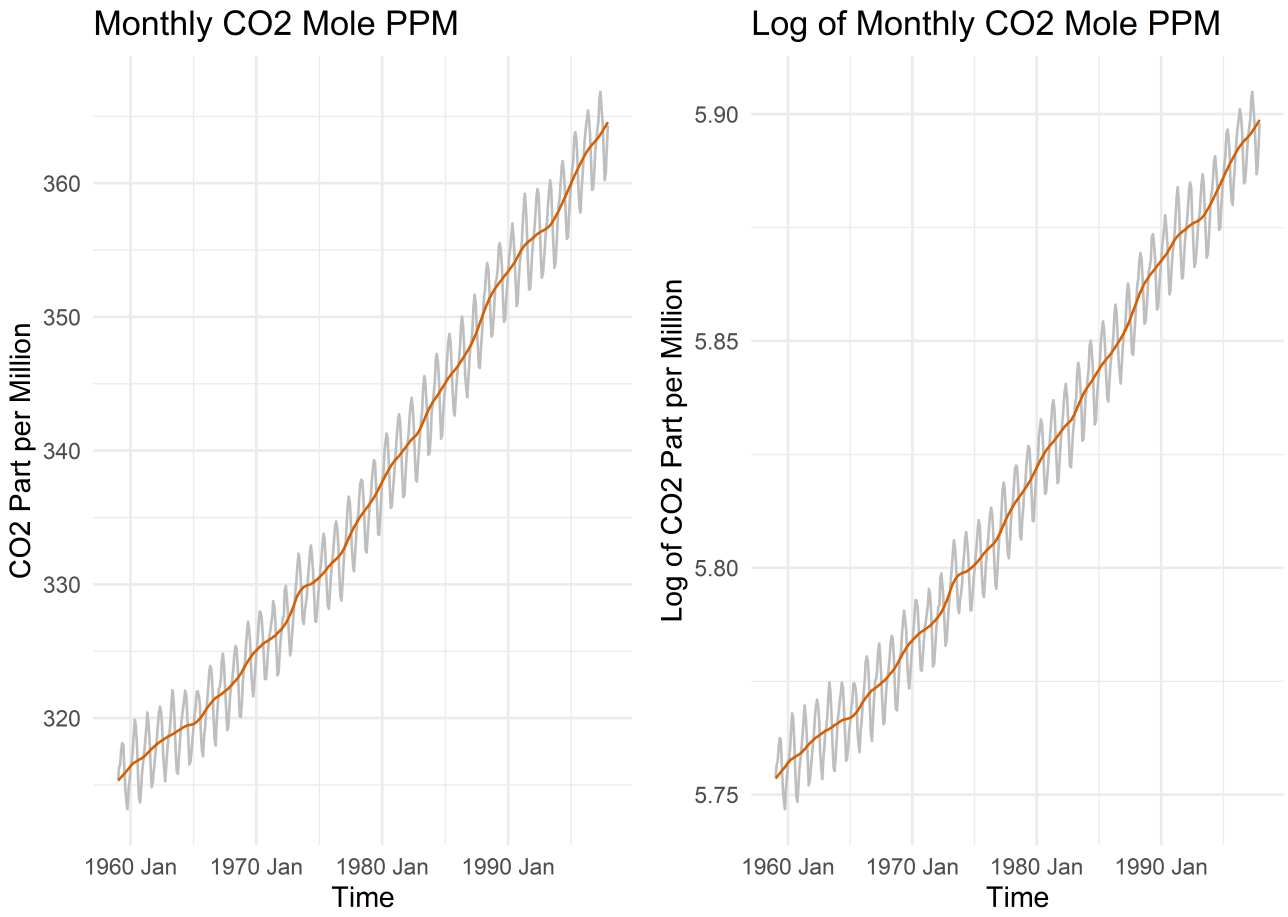
Background

The data is the accumulation of carbon dioxide in the Earth's atmosphere based on continuous measurements taken at the Mauna Loa Observatory on the island of Hawaii from 1959 to the present day. In addition to the summit's lack of vegatation shielding it from the effects of local trends in CO2 emmissions, the altitude (3,400 meter) of the site is well situated to measure representative air masses for a large area. Furthermore, Keeling and his collaborators measured the incoming ocean breeze above the thermal inversion layer to

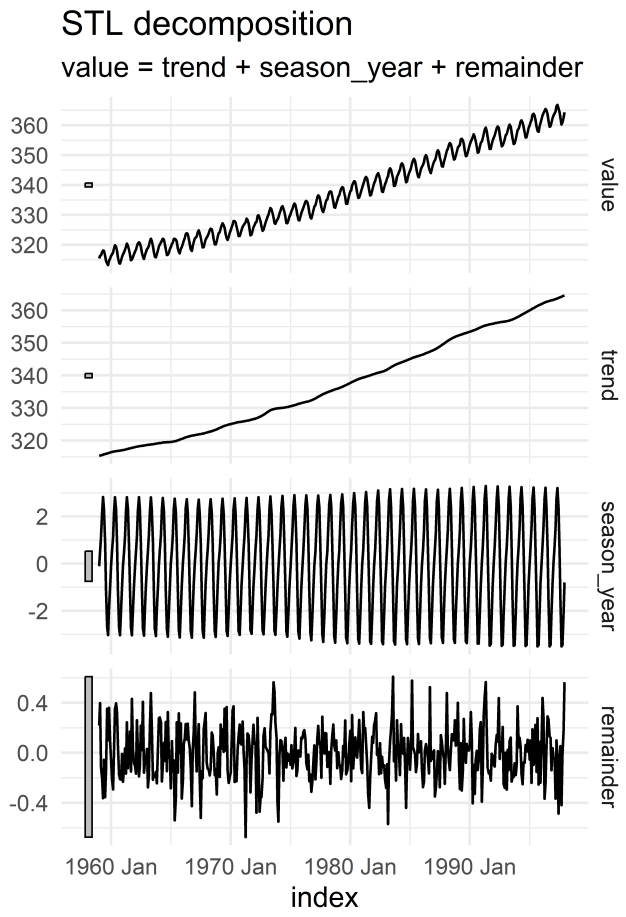
minimize local contamination from volcanic vents. The data is normalized to remove any influence from local contamination.

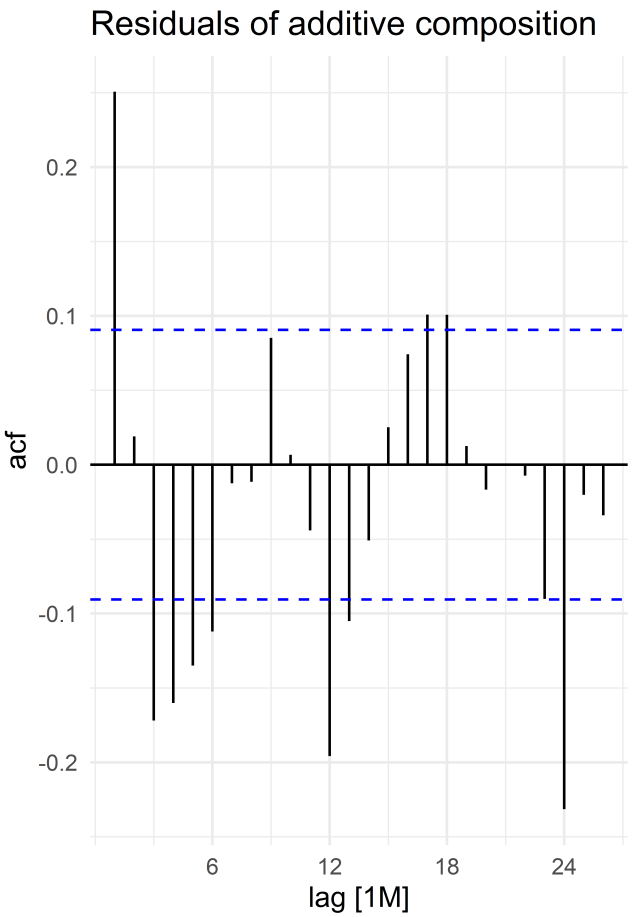
The CO2 data is collected by a CO2 analyzer that uses a technique called “Cavity Ring-Down Spectroscopy (CRDS). The measurements are made with an infrared spectrophotometer known as a non-dispersive infrared sensor, which is calibrated using World Meteorological Organization standards. The site has used the same sensor type since 1959. In addition, the CO2 measurements are compared with other individual measurements to ensure the accuracy of the measured data.

Exploratory Data Analysis (EDA)



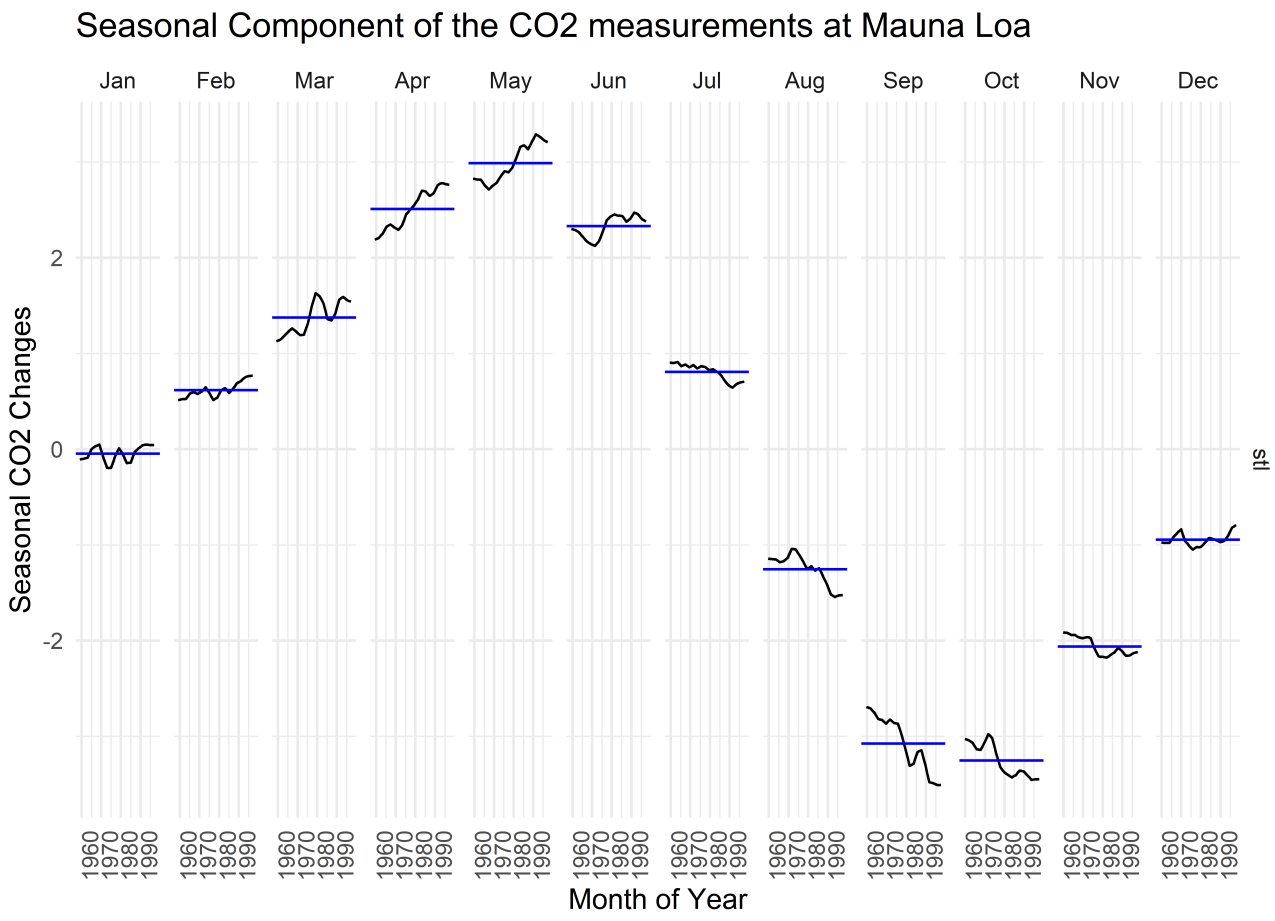
The team performed exploratory data analysis (EDA) to understand the timeseries data of CO2 mole fraction (part per million - ppm) at Mauna Loa. In the above, the left plot shows the trend between the CO2 measurements with time and on the right the logarithm of the CO2 measurements with time. The trend lines were generated using additive (left) and multiplicative (right) decomposition methods. The left plot suggests an approximate constant growth rate, and thus an additive decomposition method may be a sufficient for this problem. However, the data was fit with both methods and used the diagnostic statistics to determine which method is a more appropriate.



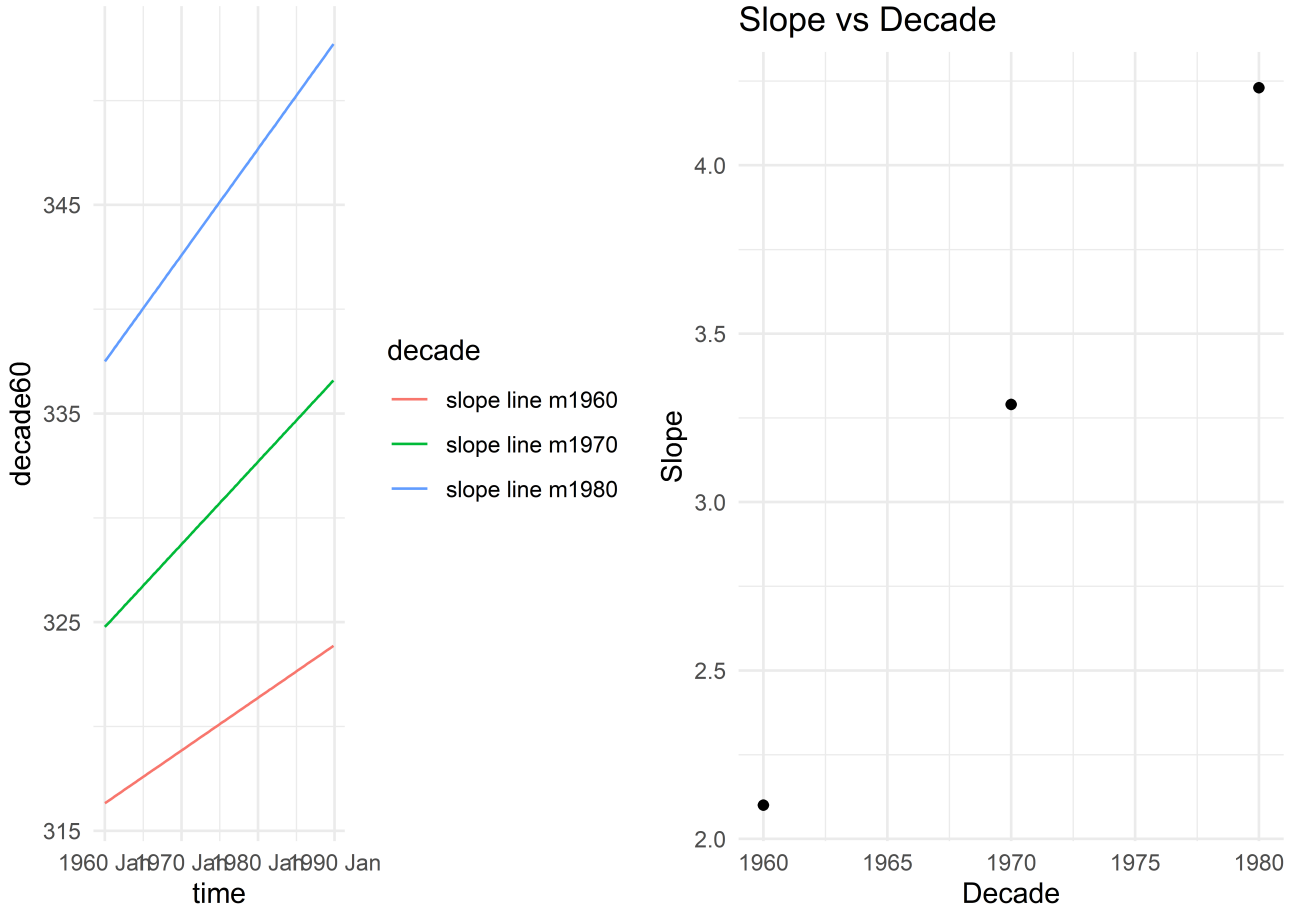


The second plot above shows the actual decomposition using the additive method, the autocorrelation (ACF) plot of the residuals. The linear trend portion suggests that from 1959 to 1997, the CO2 level increases about 40 ppms, the seasonality portion suggests that the CO2 mole fraction varies by 2 ppm depending on season. The ACF plots show that although the residuals seem to be stationary, they do not follow a white noise signal (there are significant lags within the plot), suggesting a linear time and seasonal model may not be sufficient to fit the data. The plots of the multiplicative method show the the same results, therefore they are omitted.

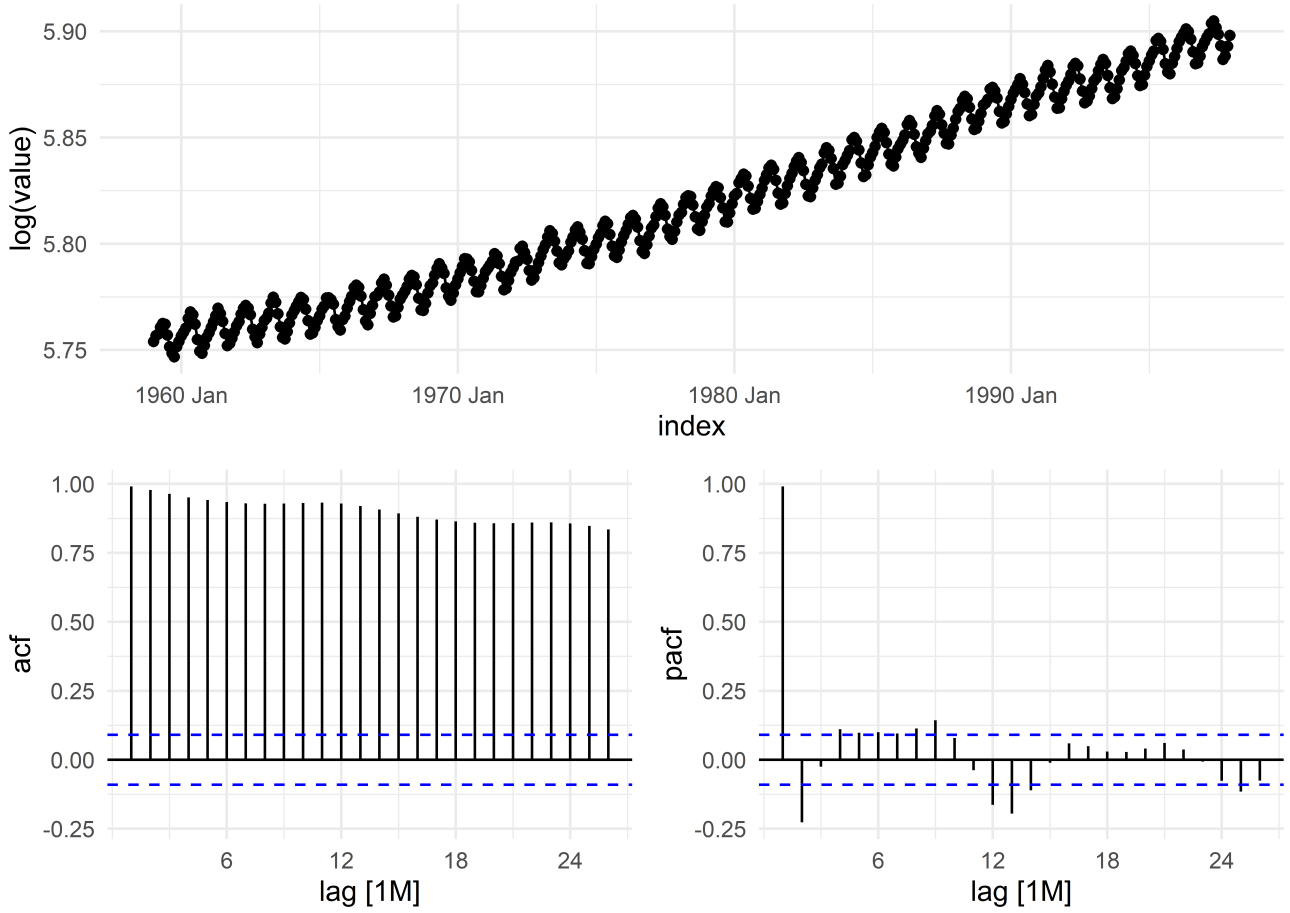
As noted above the seasonal component appears to have a mode and antimode of -2 ppm to 2 ppm, with a slight increase in amplitude over time. This increase is an interesting finding that we will explore next.



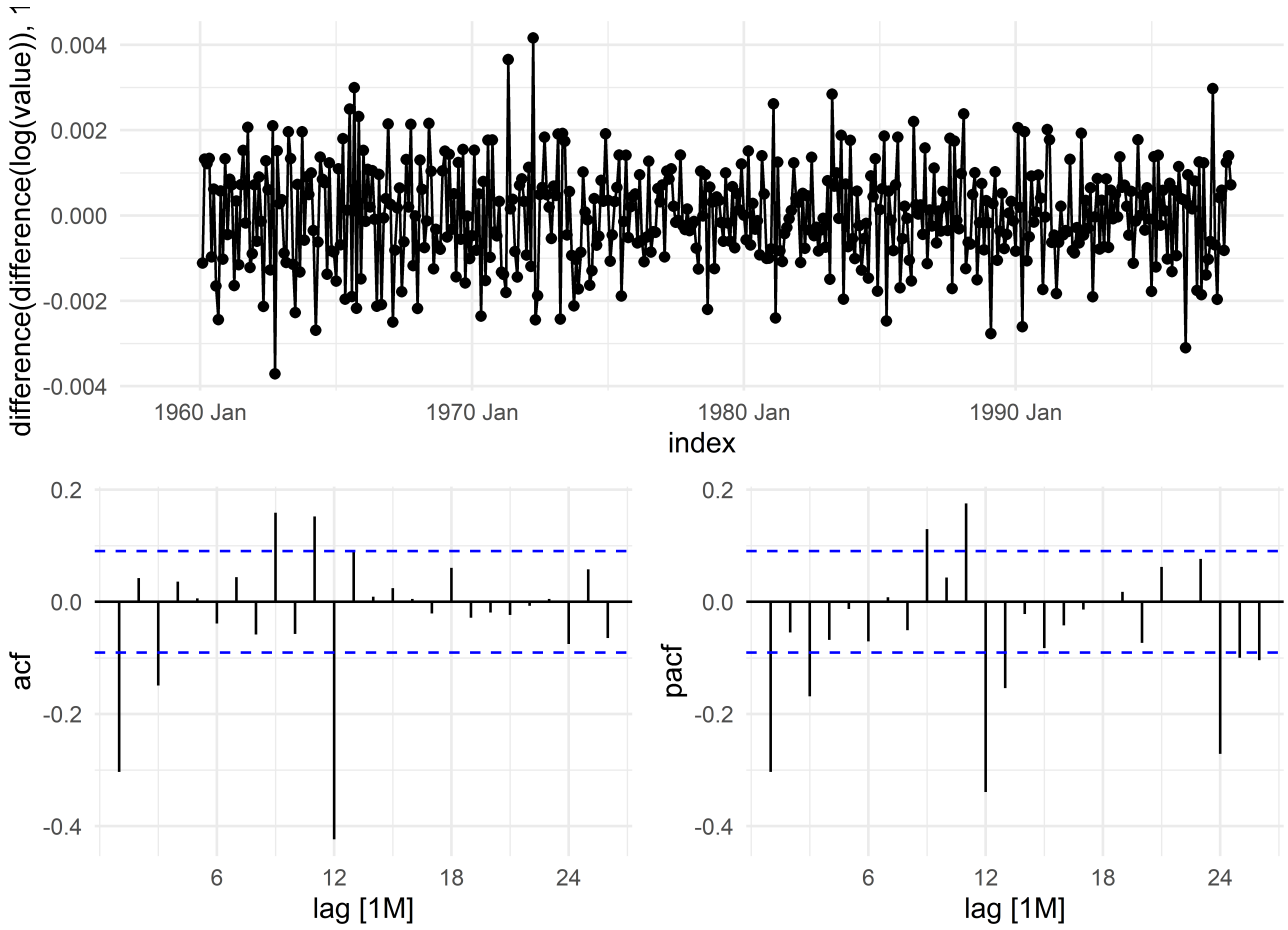
A seasonal subseries plot (above) facets the time series by each season in the seasonal period. This function is particularly useful in identifying changes in the seasonal pattern over time. From the plot above we can see that from 1960 to 1990 the mode and antimode have increased over the years, this is particulary visible in the month of May, August, September and October. We will explore if this increase in amplitude affects the overall growth trend.



The plot on the left is each decade fitted to a linear model. We have complete decades from 1960 up to the decade of 1980. From the plot we can see that the slope is increasing. On the right is the scatter plot of the slope for each decade. The rate of growth increased from an average of 2.2 PPM to 4.4 PPM over three decades. We expect an exponential model to better fit the data. The CO2 trend not only includes a seasonal component with an increasing amplitude but also an increasing slope over time. It would be satisfying to find out if those two components are correlated. That is beyond the scope of this analysis.



The first plot shows the ACF and PACF plots of the raw data after its log transformation. The ACF portion of this plot shows a strong trend coupling with a seasonality signal.



The team performed a series of transformation steps to remove the trend and seasonality component from the CO2 timeseries data. The team took a difference of the CO2 measurements in order to remove the trend component. The ACF and PACF portion of the first order differencing of the CO2 measurements showed that a seasonality component remained at every 12 lags and the ACF portion was very distinct from white noise. To remove the seasonality component, the team performed a further differencing at every 12 lags of the dataset. The plot above shows that the dataset after two differencing is closer to random white noise. The PACF portion shows lag 1 and lag 12 are significant and the ACF portion shows that lag 1 and 12 are significant .This suggests that we can use a ma1 and seasonal ma1, and an ar1 and seasonal ar1 model to fit the data. This information will be used when fitting the ARIMA model. The team performed the analysis on the additive and multiplicative model, but we are showing the results for the multiplicative model only, as the additive model showed to have higher Akaike information criterion (AIC) score.

Linear time trend model

Fitting the Data to an LTTM

```
add.mdls <- df %>%
  model(add_lin = TSLM(value ~ trend()),
        add_quad = TSLM(value ~ trend() + I(trend()^2)),
        add_lin_sea = TSLM(value ~ trend() + season()),
        add_quad_sea = TSLM(value ~ trend() + I(trend()^2) + season()))

add.mdls %>% report()
```

```
## # A tibble: 4 × 15
##   .model    r_squared adj_r_squared sigma2 statistic p_value    df log_lik   AIC
##   <chr>         <dbl>         <dbl>  <dbl>     <dbl>   <dbl> <int>   <dbl> <dbl>
## 1 add_lin       0.969         0.969  6.85     14795.       0     2  -1113.  905.
## 2 add_quad      0.979         0.979  4.76     10750.       0     3  -1028.  735.
## 3 add_lin_…     0.988         0.988  2.68      3218.       0    13   -888.  476.
## 4 add_quad…     0.998         0.998  0.524    15315.       0    14   -506. -286.
## # ℹ 6 more variables: AICc <dbl>, BIC <dbl>, CV <dbl>, deviance <dbl>,
## #   df.residual <int>, rank <int>
```

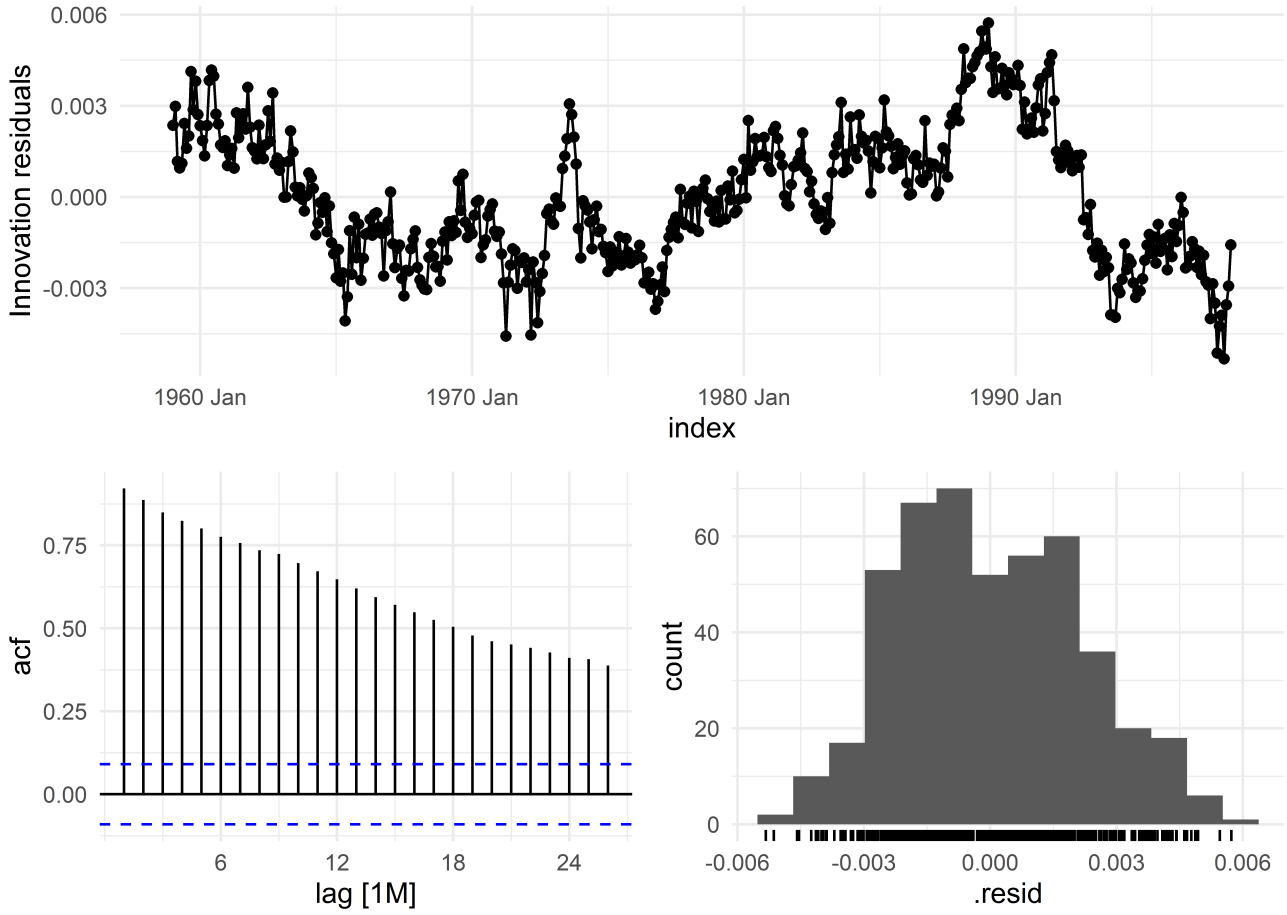
```
mul.mdls <- df %>%
  model(mul_lin = TSLM(log_value ~ trend()),
        mul_quad = TSLM(log_value ~ trend() + I(trend()^2)),
        mul_lin_sea = TSLM(log_value ~ trend() + season()),
        mul_quad_sea = TSLM(log_value ~ trend() + I(trend()^2) + season()))
mul.mdls %>% report()
```

```
## # A tibble: 4 × 15
##   .model  r_squared adj_r_squared  sigma2 statistic p_value    df log_lik    AIC
##   <chr>       <dbl>         <dbl>   <dbl>     <dbl>   <dbl> <int>   <dbl>  <dbl>
## 1 mul_lin     0.972         0.972 5.44e-5    16325.       0     2   1635. -4591.
## 2 mul_qu…     0.979         0.978 4.21e-5    10609.       0     3   1695. -4710.
## 3 mul_li…     0.991         0.991 1.75e-5     4316.       0    13   1906. -5112.
## 4 mul_qu…     0.998         0.998 4.82e-6    14529.       0    14   2208. -5713.
## # ℹ 6 more variables: AICc <dbl>, BIC <dbl>, CV <dbl>, deviance <dbl>,
## #   df.residual <int>, rank <int>
```

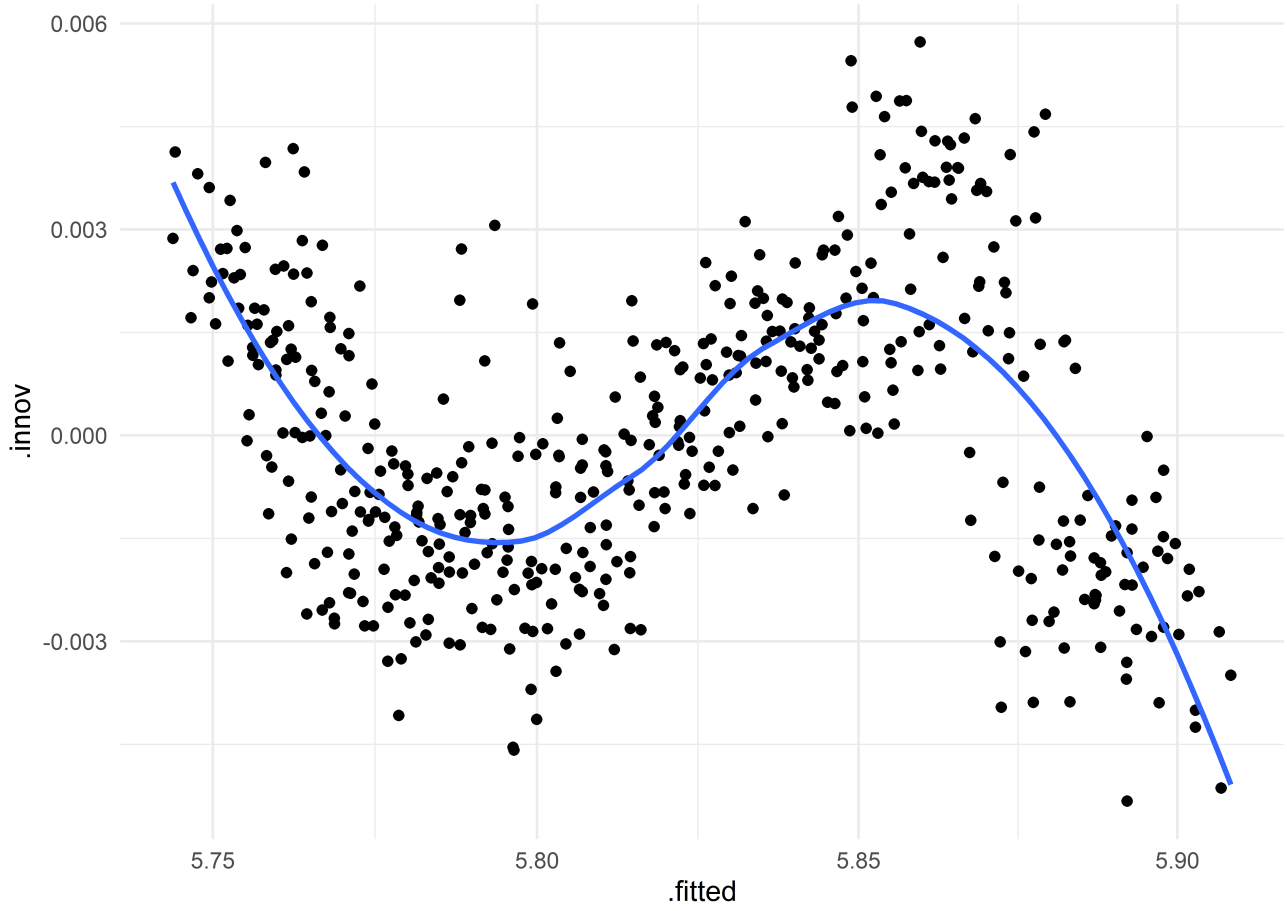
```
mul.quad.sea <- df %>%
  model(trend_model = TSLM(log_value ~ trend() + I(trend()^2) + season()))
mul.quad.sea %>% report()
```

```
## Series: log_value
## Model: TSLM
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -0.0053270 -0.0017362 -0.0001774  0.0015139  0.0057292
##
## Coefficients:
##                   Estimate Std. Error   t value Pr(>|t|)
## (Intercept)     5.751e+00  4.533e-04 12687.967  < 2e-16 ***
## trend()         2.223e-04  3.012e-06    73.817  < 2e-16 ***
## I(trend()^2)    2.150e-07  6.219e-09    34.566  < 2e-16 ***
## season()year2   1.969e-03  4.974e-04     3.959 8.73e-05 ***
## season()year3   4.163e-03  4.974e-04     8.371 7.16e-16 ***
## season()year4   7.498e-03  4.974e-04    15.075  < 2e-16 ***
## season()year5   8.911e-03  4.974e-04    17.916  < 2e-16 ***
## season()year6   6.965e-03  4.974e-04    14.004  < 2e-16 ***
## season()year7   2.480e-03  4.974e-04     4.986 8.78e-07 ***
## season()year8  -3.662e-03  4.974e-04    -7.362 8.61e-13 ***
## season()year9  -9.098e-03  4.974e-04   -18.290  < 2e-16 ***
## season()year10 -9.661e-03  4.974e-04   -19.423  < 2e-16 ***
## season()year11 -6.113e-03  4.974e-04   -12.290  < 2e-16 ***
## season()year12 -2.799e-03  4.974e-04    -5.627 3.21e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.002196 on 454 degrees of freedom
## Multiple R-squared: 0.9976,  Adjusted R-squared: 0.9975
## F-statistic: 1.453e+04 on 13 and 454 DF, p-value: < 2.22e-16
```

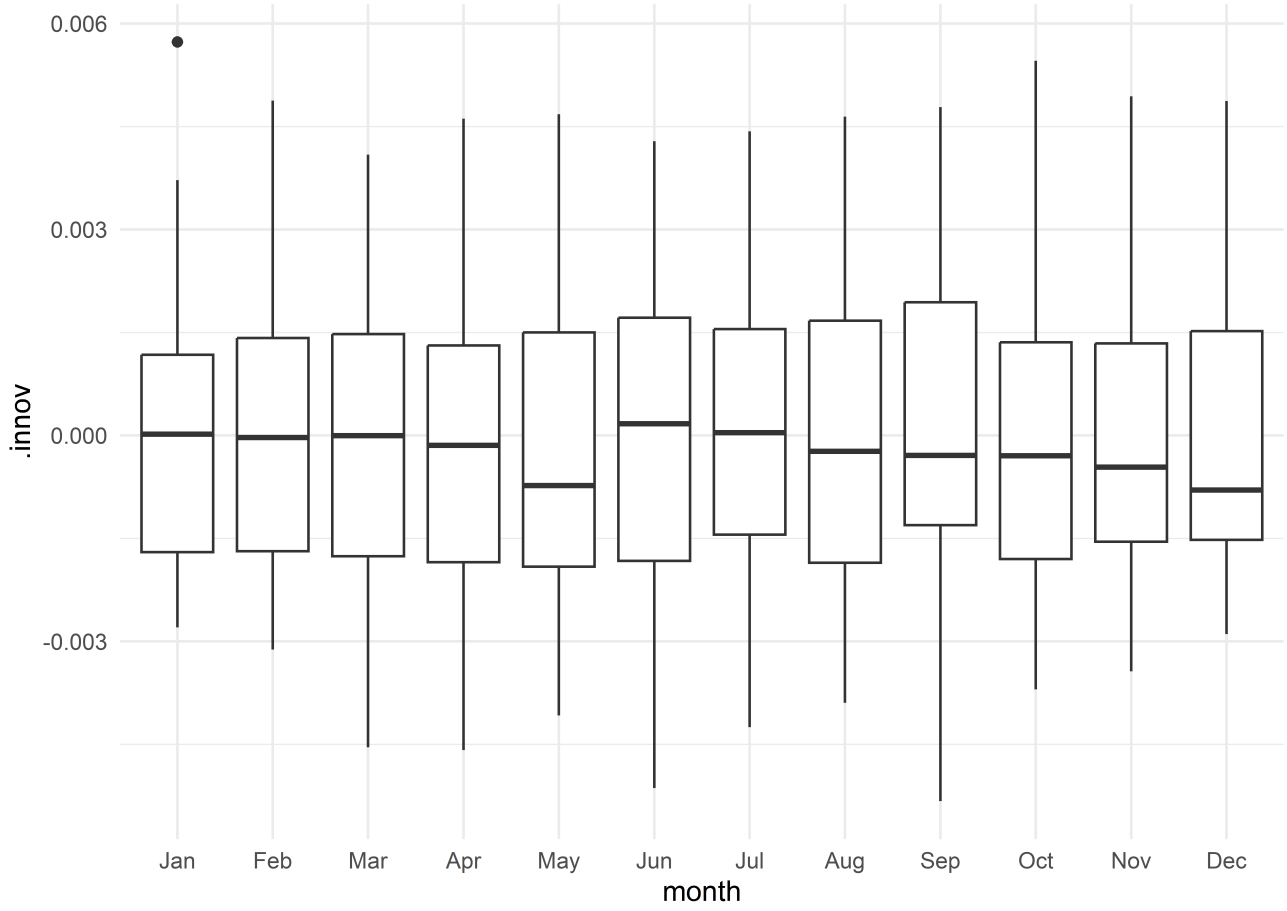
The team fit the data using four modeling combinations: linear trend only, linear with quadratic trend, linear trend with seasonal, and linear trend with quadratic trend and seasonal. For each modeling combination, the team used both decomposition methods to fit the data. The team selected the model with the lowest AIC score. Based on the information presented in the above tables, and as expected, the model using multiplicative decomposition methodology and including the linear trend, quadratic trend and seasonal terms is the best model out of the eight models. The summary table shows that all coefficients were statistically different from 0. The adjusted R square of the model is 0.9976, which indicates that more than 99% of the response variance is explained in this model.




```
## `geom_smooth()` using method = 'loess' and formula = 'y ~ x'
```



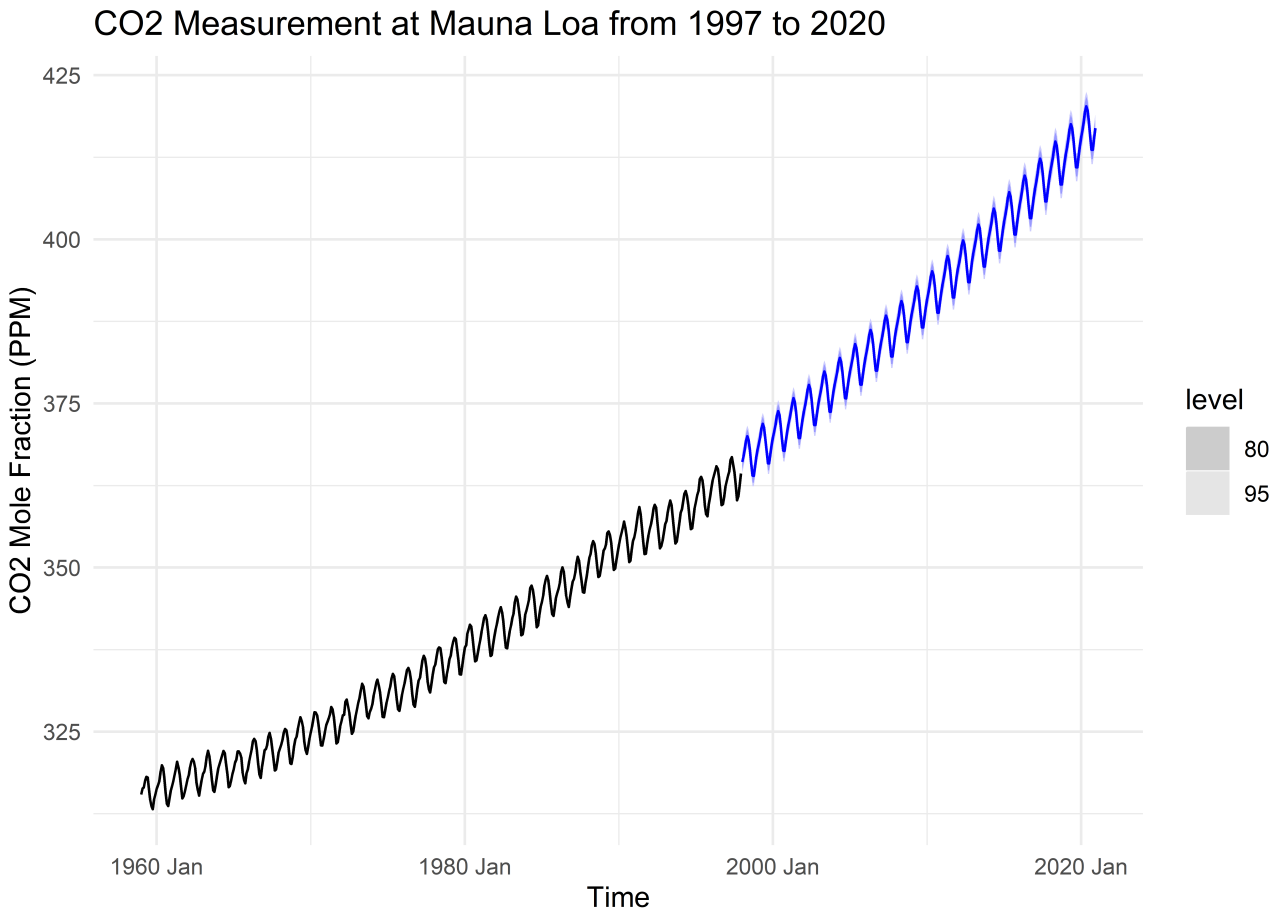
The team also conducted a residual analysis on the selected LTTM. The residual ACF plot shows that the residuals do not follow a white noise signal and is not a stationary timeseries. The residual against fitted value plot also shows that there is a relationship between the model residuals and the fitted value, which indicates that there are confounding variables.




```
## # A tibble: 1 × 3
##   .model      lb_stat lb_pvalue
##   <chr>         <dbl>     <dbl>
## 1 trend_model  10741.         0
```

Moreover, the ljung-box test (p-value less than 0.05) also shows that the residuals of the model exhibit a serial correlation. The diagnostics indicate that structure remains in the residual data, however a model with such residuals can be tolerated if the model provides useful forecasts. The team decided not to add more variables because of concerns with overfitting and as a consequence poor forecasting.

Forecasting with a LTT model




```
## # A tsibble: 6 x 5 [1M]
## # Key:       .model [1]
##   .model         index      log_value .mean                 `90%`
##   <chr>          <mth>         <dist> <dbl>                <hilo>
## 1 trend_model 2020 Jul lN(6, 7.3e-06)  418. [416.1660, 419.8688]90
## 2 trend_model 2020 Aug lN(6, 7.3e-06)  416. [413.8391, 417.5258]90
## 3 trend_model 2020 Sep lN(6, 7.3e-06)  414. [411.8158, 415.4893]90
## 4 trend_model 2020 Oct lN(6, 7.3e-06)  414. [411.8042, 415.4823]90
## 5 trend_model 2020 Nov lN(6, 7.3e-06)  415. [413.4894, 417.1873]90
## 6 trend_model 2020 Dec lN(6, 7.3e-06)  417. [415.0845, 418.8014]90
```

The team used the selected model to generate CO2 mole fraction forecasts from December 1997 to December 2020 (as shown in the above graph). The generated forecast indicates a similar pattern of growth for the CO2 level in the atmosphere. By 2020, the CO2 mole fraction is predicted to reach approximately 417 ppm with a 90% confidence interval (CI) between 415 and 419 ppm.

ARIMA times series model

Fitting the Data to an ARIMA model

We will now choose an ARIMA model to fit to the time series.

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

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

## # A tibble: 1 × 2
##   kpss_stat kpss_pvalue
##       <dbl>       <dbl>
## 1    0.0115         0.1
```

Following the initial analysis done in the EDA section, the team used a unit root test to determine the number of differences (both seasonal and non-seasonal) to make the CO2 timeseries stationary. The unit root test results show that it requires one seasonal and one non-seasonal difference to make the timeseries stationary. These results are in agreement with our findings in the explanatory data analysis section.

The KPSS test of the CO2 timeseries after applying the differences (at lag 1 and lag 12) shows a p-value of 0.1, which is greater than 0.05, therefore we fail to reject the null hypothesis that the time series is stationary after two differencing.

The ACF portion in the EDA showed that we can expect to use multiple ma1 non-seasonal terms and a seasonal MA1 model to fit the data, the EDA did not indicate a need for an AR term.

```
ari.fit <- df %>%
  model(ARIMA111111 = ARIMA(log(value) ~ pdq(1,1,1) + PDQ(1,1,1)),
        ARIMA111211 = ARIMA(log(value)  ~ pdq(1,1,1) + PDQ(2,1,1)),
        ARIMA211211 = ARIMA(log(value)  ~ pdq(2,1,1) + PDQ(2,1,1)),
        ARIMA112112 = ARIMA(log(value)  ~ pdq(1,1,2) + PDQ(1,1,2)),
        ARIMA112111 = ARIMA(log(value)  ~ pdq(1,1,2) + PDQ(1,1,1)),
        ARIMA111112 = ARIMA(log(value)  ~ pdq(1,1,1) + PDQ(1,1,2)),
        ARIMA013011 = ARIMA(log(value)  ~ pdq(0,1,3) + PDQ(0,1,1)),
        auto = ARIMA(log(value), stepwise = FALSE, approx = FALSE)
        )
```

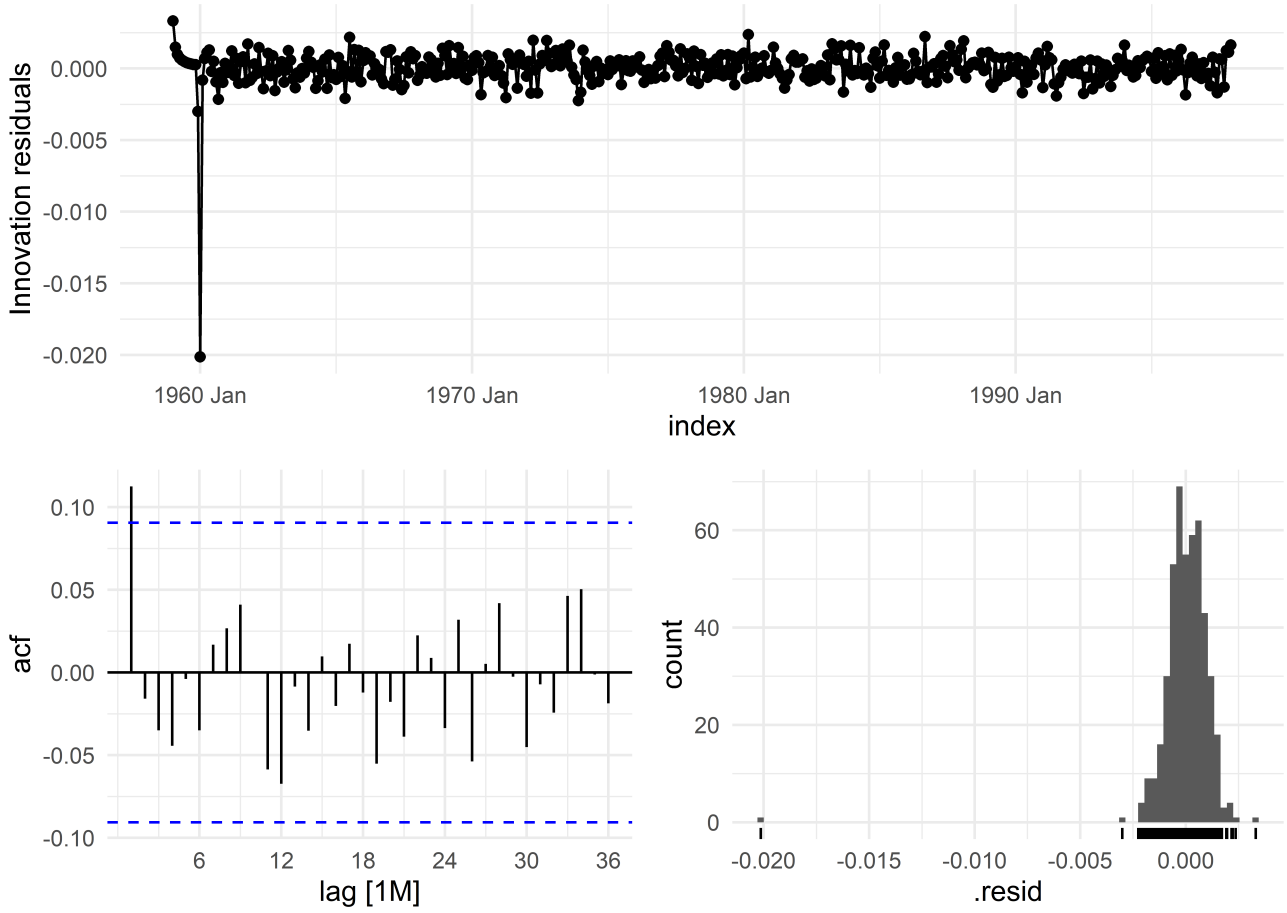
```
ari.fit %>% pivot_longer(everything(), names_to = "Model name", values_to = "Orders")
```

```
## # A mable: 8 x 2
## # Key:     Model name [8]
##   `Model name`                    Orders
##   <chr>                          <model>
## 1 ARIMA111111  <ARIMA(1,1,1)(1,1,1)[12]>
## 2 ARIMA111211  <ARIMA(1,1,1)(2,1,1)[12]>
## 3 ARIMA211211  <ARIMA(2,1,1)(2,1,1)[12]>
## 4 ARIMA112112  <ARIMA(1,1,2)(1,1,2)[12]>
## 5 ARIMA112111  <ARIMA(1,1,2)(1,1,1)[12]>
## 6 ARIMA111112  <ARIMA(1,1,1)(1,1,2)[12]>
## 7 ARIMA013011  <ARIMA(0,1,3)(0,1,1)[12]>
## 8 auto         <ARIMA(1,1,0)(2,1,2)[12]>
```

```
glance(ari.fit) |> arrange(AIC) |> select(.model:AIC)
```

```
## # A tibble: 8 × 4
##   .model          sigma2 log_lik    AIC
##   <chr>            <dbl>   <dbl>  <dbl>
## 1 ARIMA013011 0.00000165   2572. -5133.
## 2 ARIMA111211 0.00000165   2571. -5131.
## 3 ARIMA211211 0.00000165   2572. -5131.
## 4 ARIMA111111 0.00000165   2570. -5130.
## 5 ARIMA112111 0.00000165   2571. -5130.
## 6 ARIMA111112 0.00000165   2571. -5130.
## 7 ARIMA112112 0.00000166   2571. -5129.
## 8 auto        0.00000166   2569. -5125.
```

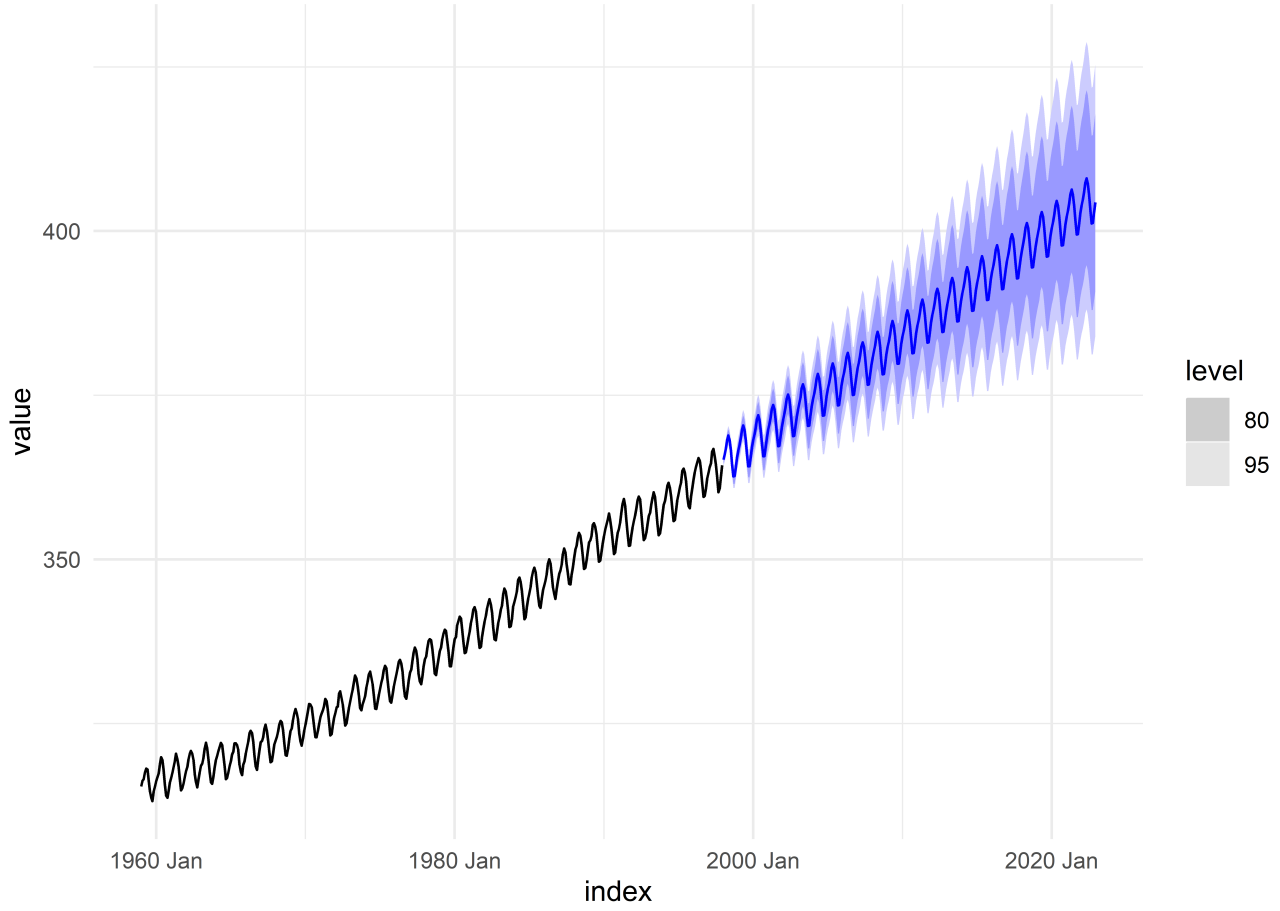
Based on the results of testing out different model combination, the SARIMA model with one seasonal and one non-seasonal differencing, three non-seasonal moving average terms and one seasonal moving average term seems to be the model with the best performance (lowest AIC), which is slightly different from our EDA analysis.




```
## # A tibble: 8 × 3
##   .model      lb_stat lb_pvalue
##   <chr>         <dbl>     <dbl>
## 1 ARIMA013011    122.      1.00
## 2 ARIMA111111    124.      1.00
## 3 ARIMA111112    124.      1.00
## 4 ARIMA111211    122.      1.00
## 5 ARIMA112111    125.      1.00
## 6 ARIMA112112    124.      1.00
## 7 ARIMA211211    123.      1.00
## 8 auto           122.      1.00
```

The residual analysis of the selected model shows that the residuals timeseries is white noise signal. The ljung-box test result shows that all selected models have p-value greater than 0.05, which suggests that the residuals of the selected model is stationary. The ARIMA model accounts for the structure of the data better than our LTTM model. ARIMA models are more flexible then LTTM as they can capture more complex patterns in the data.

Forecasting with an ARIMA model

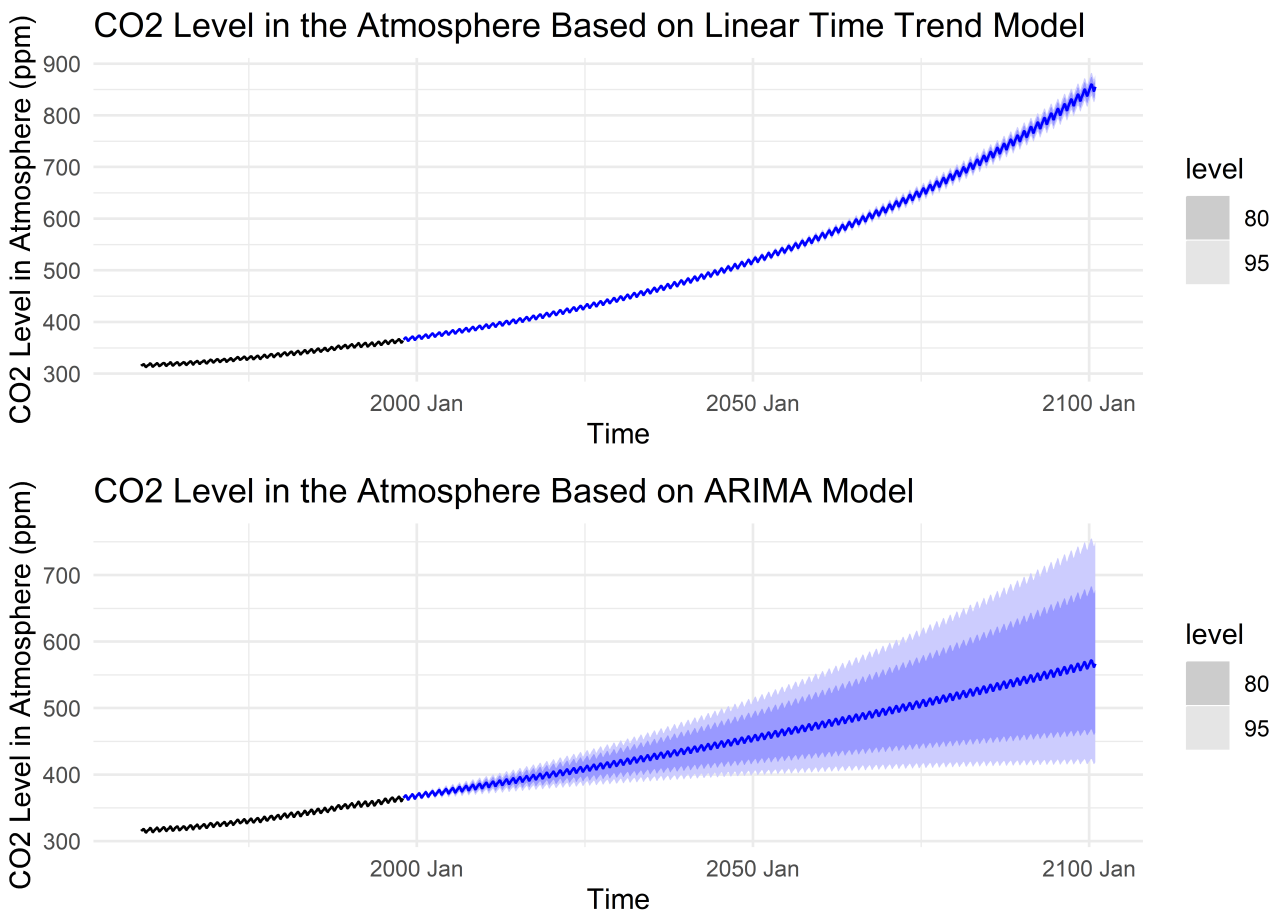



```
## # A tsibble: 6 x 5 [1M]
## # Key:       .model [1]
##   .model         index            value .mean                  `90%`
##   <chr>          <mth>           <dist> <dbl>                 <hilo>
## 1 ARIMA013011 2022 Jul t(N(6, 0.00066))  406. [388.7085, 422.9186]90
## 2 ARIMA013011 2022 Aug t(N(6, 0.00066))  403. [386.4154, 420.5499]90
## 3 ARIMA013011 2022 Sep t(N(6, 0.00067))  401. [384.3505, 418.4278]90
## 4 ARIMA013011 2022 Oct t(N(6, 0.00067))  401. [384.3485, 418.5505]90
## 5 ARIMA013011 2022 Nov t(N(6, 0.00068))  403. [385.8291, 420.2877]90
## 6 ARIMA013011 2022 Dec t(N(6, 0.00068))  404. [387.2602, 421.9715]90
```

The team used the selected model to generate CO2 level in the atmosphere to December 2022. Based on the generated forecast, in December 2022, the forecasted CO2 level will be approximately 404 ppm with a 90% CI between 387 and 422 ppm. This result is lower than the forecast using the LTTM model which already forecasts 417 ppm by 2020, two years before.

Forecast atmospheric CO2 growth

Comparing LTTM and ARIMA Forecasts



The team generated forecast from 1998 to 2100. Based of the generated forecast, for the linear model, the CO2 level at Mauna Loa is expected to reach 420 ppm the first time in 2020 May and last time in 2022 Nov. However, based on the 90% CI band, we may observe the CO2 level to reach 420 ppm as early as in 2020 Apr and may see it again the last time in 2023 Oct. Also, according to the linear model, the CO2 level at Mauna Loa is expected to reach 500 ppm the first time in 2045 Mar and last time in 2046 Oct. But based on its 90% CI, we may observe the CO2 level to reach 500 ppm as early as 2044 Mar and may see it again the last time in 2047 Oct.

For the ARIMA model, the CO2 level is expected to reach 420 ppm the first time in 2029 May and last time in 2033 Oct. Based on its 90% CI, we may observe the CO2 level to reach 420 ppm as early as in 2020 May and may see it again the last time in 2070 Oct. The CO2 level is expected to reach 500 ppm the first time in 2070 Apr and last time in 2073 Oct. Based on its 90% CI, we may observe the CO2 level to reach 500 ppm as early as 2044 Mar and may come back again to this level after the last time step of the forecast horizon, which is 2100 Dec.

In 2100, based on the linear model, the CO2 level at Mauno Loa is expected to reach 853.3740519 ppm for the linear model and 567.1541446 ppm for the ARIMA model. The ARIMA model suggests a slower growth of CO2 level and also has a wider 90% CI, than the predictions from the linear model.

Athough it is unlikely both models would provide accurate CO2 levels in 2100 (given they do not account for changing external trends), one model can be closer in its prediction than the other model. ARIMA models are stochastic models and by their essence are generally inadequate for long-term forecasting, such as more than a few months ahead. This fact is reflected by its growing band of confidence interval with time. The LTTM is a deterministic model that depends heavily on an underlying process and if the process generating the data changes, the model will continue to forecast based on the original process. If the underlying reason for CO2 emissions is deterministic, then our team does not reasonably expect the trend to change drastically and thus we would have more confidence in the LTTM prediction. We can only determine which model is better, and also, if the CO2 emissions trend is a stochastic or deterministic, by measuring their forecasting. The statical diagnostics in our analysis lean towards favoring the ARIMA model and hence a stochastic trend.

Report from the Point of View of the Present

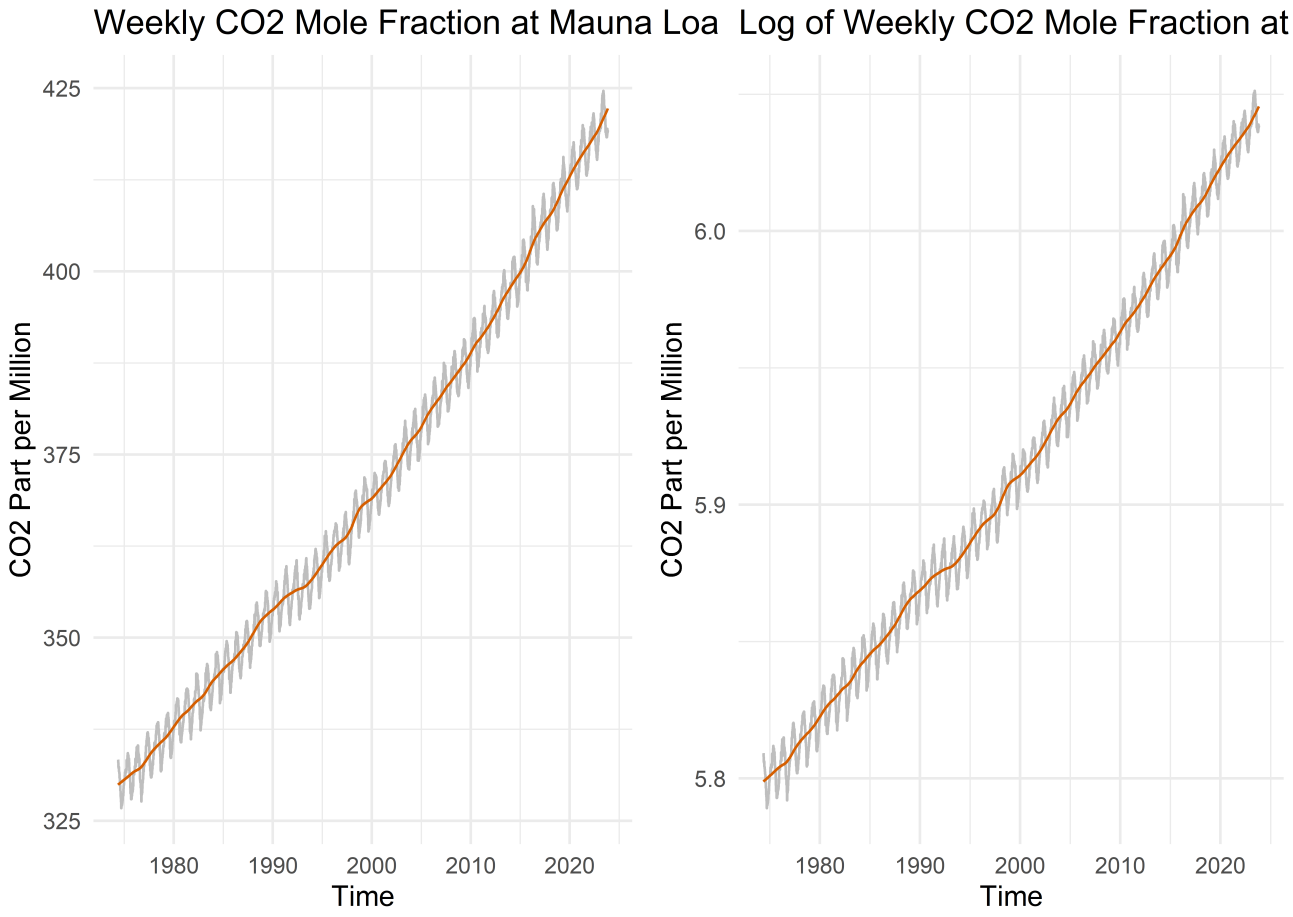
Introduction

Having journeyed through the historical perspective presented in our 1997 report, we now stand at the present, armed with updated data and refined tools. However, the severity of CO2 levels in the near future is not known with certainty due to the uncertainty that accompanies long-term forecasts, especially those extending to the year 2100. In the following sections, the team decided to reevaluate the CO2 forecast provided in 1997 and update our forecast conducted in 1997 with more recent data. In the process of this work, we also aim to identify any systematic changes in the way CO2 level grows at Mauna Loa.

Modern data Exploration

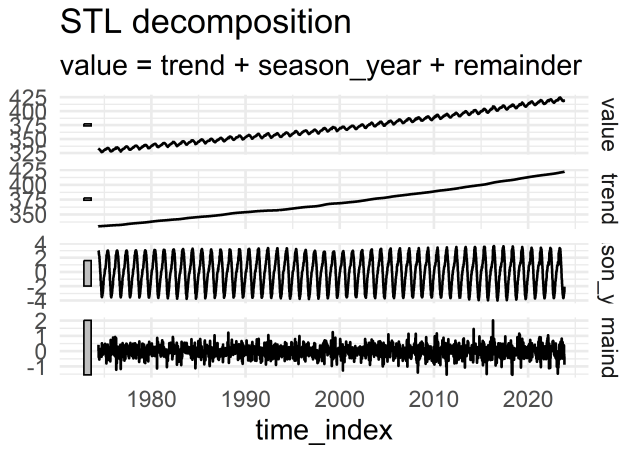
With the backdrop of our historical analysis, we turn our attention to the contemporary state of CO2 levels at Mauna Loa. Employing a methodology similar to our 1997 study, our team conducted EDA of the most recent data. This process involved meticulous steps to ensure the integrity of the data, starting with the removal of faulty values—negative readings and suspicious outliers.

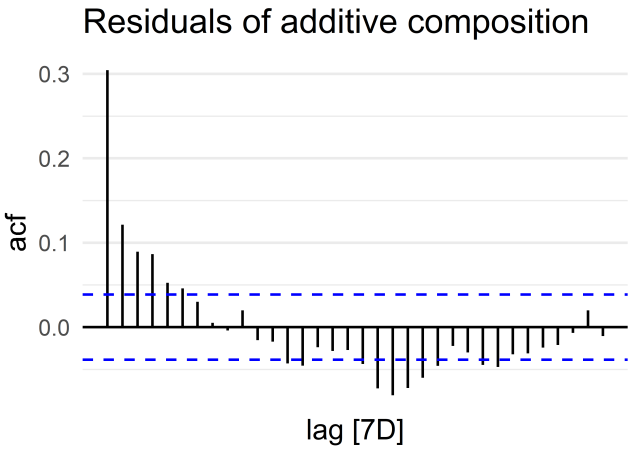
Unraveling Weekly CO2 Trends

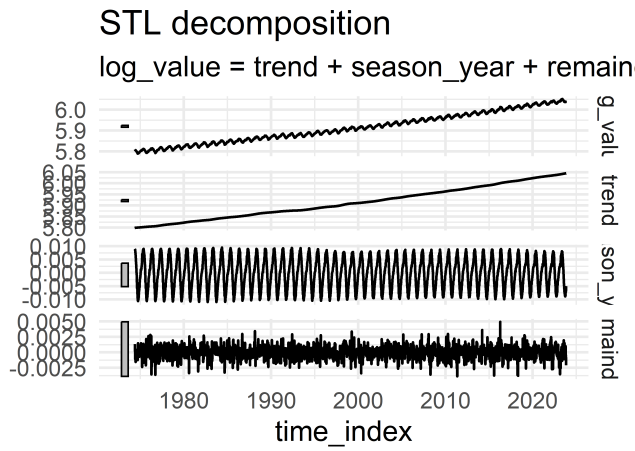


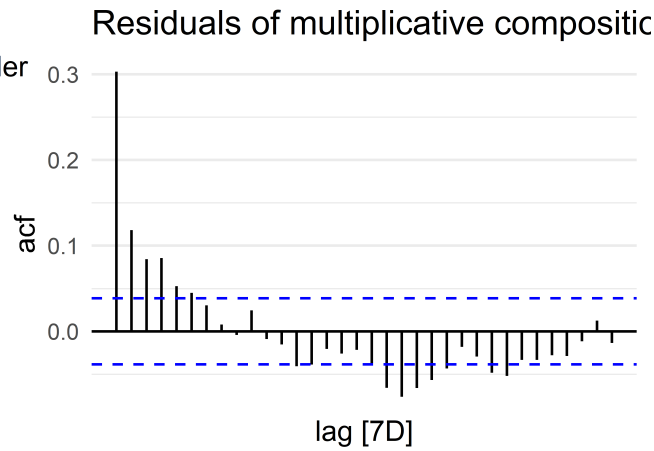
The plot above portrays the weekly CO2 levels at Mauna Loa from May 1974 to October 2023. The left graph displays the raw CO2 levels, while the right graph showcases the logarithmic transformation. Notably, the present-day plot reveals a more pronounced non-linear trend compared to the 1997 study, indicating that the multiplicative decomposition method is better suited to capture the intricacies of the current data.

Decomposition and Residual Analysis









The above plot shows the actual decomposition using the additive and multiplicative method along with the ACF plots of the residuals (after the trend and the seasonal component were removed from the series). The ACF plots show that, at the weekly granularity, the residuals timeseries does not have stationary characteristics, suggesting that further differencing transformation may be needed.

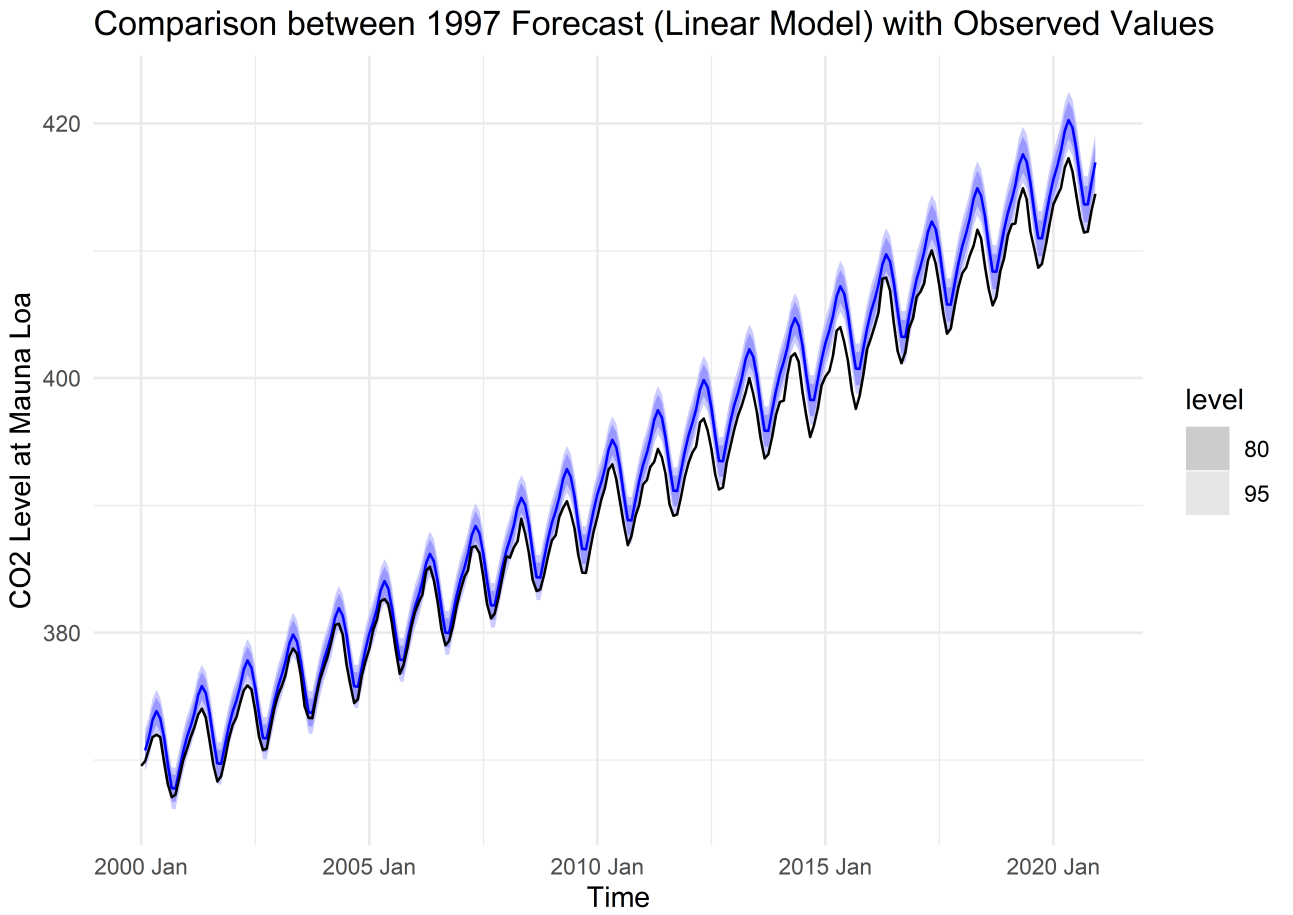
Iterative Differencing

Continuing our analysis, the team then performed a series of differencing transformation on the updated dataset. Similarly to the analysis included in the 1997 study, we generated a CO2 timeseries with the differencing at the subsequent lag and every 52 lags to remove the trend and the seasonality component from the data.

Identifying Modeling Candidates

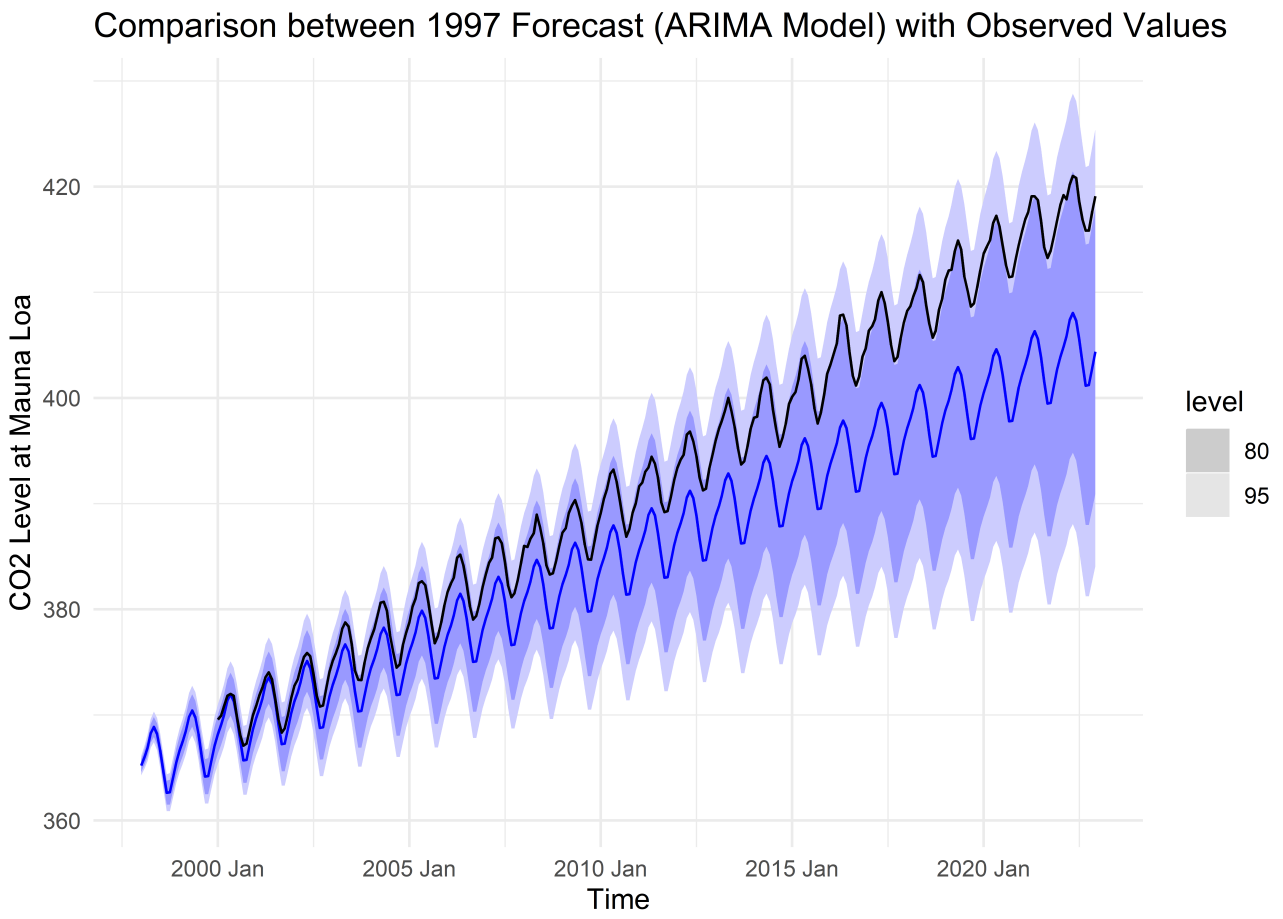
The above plots show the timeseries with differencing transformation and their associated ACF and PACF plots. The third plot, which shows the timeseries after the trend and seasonal differencing, indicates that there is no trend or seasonality within the series and suggests an ARIMA model with both seasonal and non-seasonal moving average and autoregressive terms may be a good candidate to model the updated dataset.

Linear model forecast assessment



To gauge the efficacy of our 1997 linear model in predicting present-day CO2 levelsthe forecast, we aggregated the weekly actual CO2 data to monthly data so that the dataset has the same time index as the forecast values. Based on the comparison in the above plot (where the black line shows the observed CO2 level and blue line shows the forecast CO2 level), the 1997 linear model overforecast the CO2 level at Mauna Loa from 2000 to 2020.

ARIMA model forecast assessment



The plot above depicts the comparison between the forecast from the 1997 ARIMA model and observed CO2 level. Intrestingly, the actual CO2 levels are closer to the upper bound of the 80% CI of the 1997 ARIMA forecast. The forecast from 1997 ARIMA model shows that the expected forecast is lower than the observed CO2 values at a more significant degree than that of the linear model. However, the observed CO2 still appears to be closer to the upper bound of the 80% CI of the ARIMA model.

Performance of 1997 linear and ARIMA models

The observed CO2 values indicate that the CO2 level reached 420 ppm the first time in 2022-03-27. This is earlier than the prediction from the linear model and later than the prediction from the ARIMA model. The linear model (with a tight CI band) predicted that the CO2 level would reach 420 ppm as early as in 2020 Apr, while the ARIMA model predicted that the CO2 level would reach 420 ppm as early as in 2020 May. Based on their prediction, the ARIMA model provided a more accurate expectation than the linear model.

```
## # 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 ARIMA013011 Test   6.19  7.49  6.19  1.54  1.54   4.20  4.76 0.984
```

```
## 2 ARIMA211211 Test   6.03  7.30  6.03  1.50  1.50   4.09  4.64 0.984
## 3 ARIMA111211 Test   5.93  7.19  5.93  1.48  1.48   4.02  4.57 0.984
## 4 ARIMA111111 Test   6.17  7.48  6.17  1.54  1.54   4.18  4.76 0.984
## 5 ARIMA112111 Test   6.26  7.57  6.26  1.56  1.56   4.24  4.82 0.985
## 6 trend_model Test  -1.72  1.97  1.74 -0.437 0.443  1.18  1.25 0.890
```

The team calculated the performance metrics in order to evaluate the performance of the models that we developed in 1997. The performance metrics include but are not limited to root mean square error, mean absolute error and mean absolute percentage error. Based on the results of the metrics table, the linear model with trend (linear and quadratic terms) and seasonal terms was the best model with the lowest score in all metrics.

Improved Model using New Data

Using the addition of new data and the insights we gained from evaluating 1997 modes, we can retrain the ARIMA and linear model for enhanced accuracy.

```
## # A mable: 8 x 2
## # Key:     Model name [8]
##   `Model name`                                Orders
##   <chr>                                      <model>
## 1 ARIMA111101  <ARIMA(1,1,1)(1,0,1)[12] w/ drift>
## 2 ARIMA111201  <ARIMA(1,1,1)(2,0,1)[12] w/ drift>
## 3 ARIMA211201  <ARIMA(2,1,1)(2,0,1)[12] w/ drift>
## 4 ARIMA112102                          <NULL model>
## 5 ARIMA112101                          <NULL model>
## 6 ARIMA111102  <ARIMA(1,1,1)(1,0,2)[12] w/ drift>
## 7 ARIMA013001  <ARIMA(0,1,3)(0,0,1)[12] w/ drift>
## 8 auto                  <ARIMA(0,2,2)(2,0,0)[12]>

## # A tibble: 6 × 5
##   .model           sigma2 log_lik    AIC   AICc
##   <chr>             <dbl>   <dbl>  <dbl>  <dbl>
## 1 ARIMA111201 0.000000525   3347. -6679. -6679.
## 2 ARIMA211201 0.000000526   3347. -6678. -6678.
## 3 ARIMA111102 0.000000528   3345. -6676. -6676.
## 4 ARIMA111101 0.000000532   3342. -6672. -6672.
## 5 auto        0.000000612   3323. -6636. -6636.
## 6 ARIMA013001 0.000000569   3322. -6632. -6632.

## # A tibble: 8 × 3
##   .model      lb_stat lb_pvalue
##   <chr>         <dbl>     <dbl>
## 1 ARIMA013001    258.  1.92e- 3
```

```
## 2 ARIMA111101    362.  5.74e-12
## 3 ARIMA111102    358.  1.32e-11
## 4 ARIMA111201    361.  6.88e-12
## 5 ARIMA112101     NA  NA
## 6 ARIMA112102     NA  NA
## 7 ARIMA211201    358.  1.27e-11
## 8 auto           180.  7.85e- 1
```

```
## # 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 ARIMA111201 Test     2.17e-3 2.68e-3 2.27e-3  3.60e-2 0.0375   NaN   NaN 0.746
## 2 ARIMA211201 Test     2.15e-3 2.66e-3 2.25e-3  3.56e-2 0.0372   NaN   NaN 0.746
## 3 ARIMA111102 Test     2.16e-3 2.67e-3 2.26e-3  3.58e-2 0.0374   NaN   NaN 0.751
## 4 ARIMA111101 Test     2.11e-3 2.62e-3 2.21e-3  3.50e-2 0.0366   NaN   NaN 0.755
## 5 ARIMA013001 Test     6.55e-4 1.29e-3 1.05e-3  1.08e-2 0.0174   NaN   NaN 0.581
## 6 trend_model Test    -3.58e-6 9.62e-4 8.04e-4 -6.05e-5 0.0133   NaN   NaN 0.351
```

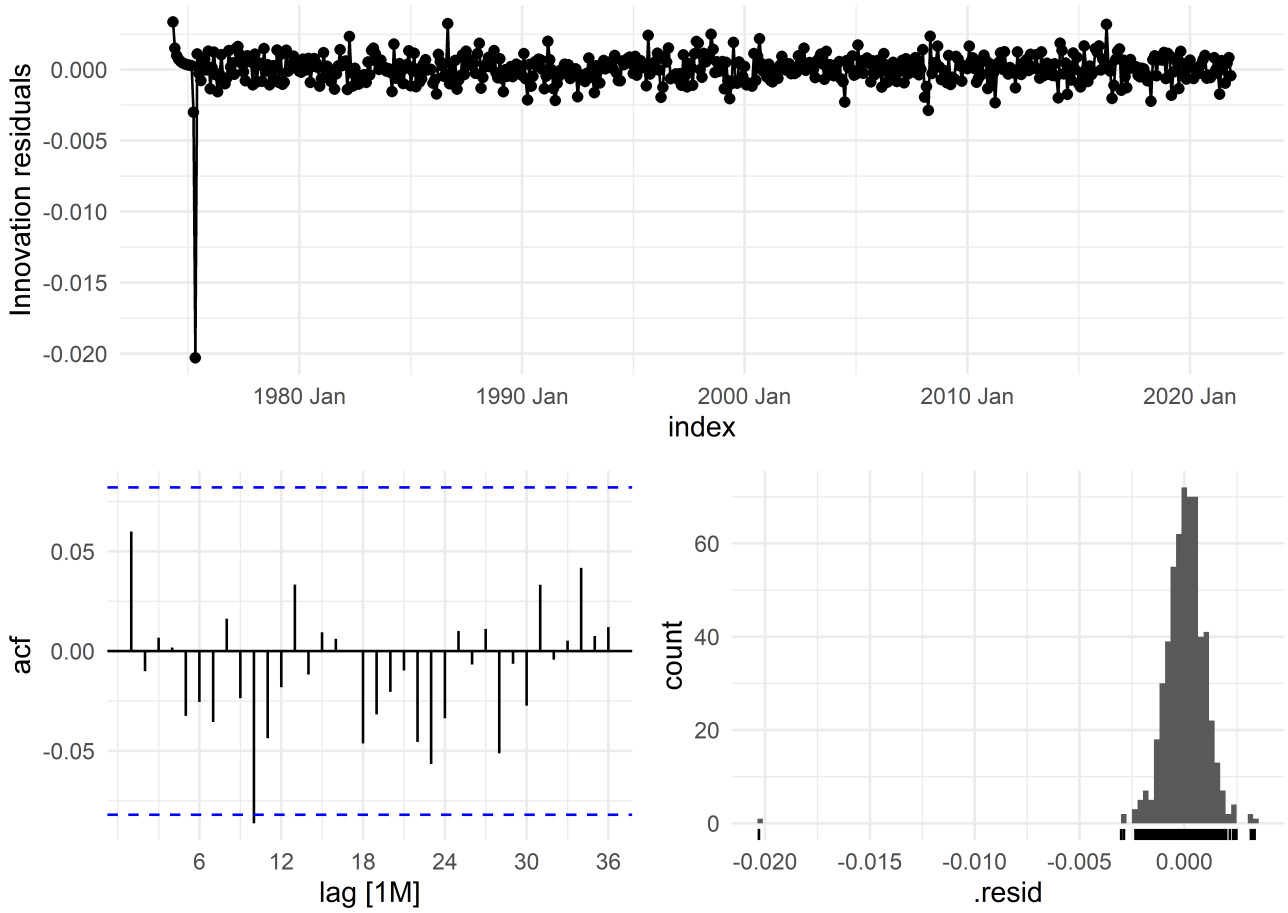
We started with developing the models using the seasonally-adjusted data (following the same process that we did in 1997). The Ljungbox test shows that all ARIMA models that use seasonally-adjusted data had residuals that show some level of serial relationship within the data. This suggests that using seasonally-adjusted data is not as effective as using non-seasonally adjusted data to develop models to fit the CO2 data and the forecast from these models may not be accurate.

```
# Non-seasonally adjusted models

ari.fit3 <- co2.log.nsa.tr %>%
  model(ARIMA011011 = ARIMA(log_value ~ pdq(0,1,1) + PDQ(0,1,1)),
        ARIMA111111 = ARIMA(log_value ~ pdq(1,1,1) + PDQ(1,1,1)),
        ARIMA111211 = ARIMA(log_value  ~ pdq(1,1,1) + PDQ(2,1,1)),
        ARIMA211211 = ARIMA(log_value  ~ pdq(2,1,1) + PDQ(2,1,1)),
        ARIMA112112 = ARIMA(log_value  ~ pdq(1,1,2) + PDQ(1,1,2)),
        ARIMA112111 = ARIMA(log_value ~ pdq(1,1,2) + PDQ(1,1,1)),
        ARIMA111112 = ARIMA(log_value  ~ pdq(1,1,1) + PDQ(1,1,2)),
        ARIMA013011 = ARIMA(log_value ~ pdq(0,1,3) + PDQ(0,1,1)),
        auto = ARIMA(log_value, stepwise = FALSE, approx = FALSE)
        )

glance(ari.fit3) |> arrange(AICc) |> select(.model:AICc)
```

```
## # A tibble: 9 × 5
##   .model          sigma2 log_lik    AIC   AICc
##   <chr>            <dbl>   <dbl>  <dbl>  <dbl>
## 1 ARIMA011011 0.00000154   3133. -6259. -6259.
```

```
## 2 auto        0.00000154   3133. -6259. -6259.
## 3 ARIMA112111 0.00000154   3135. -6258. -6257.
## 4 ARIMA111111 0.00000155   3133. -6257. -6257.
## 5 ARIMA013011 0.00000155   3133. -6256. -6255.
## 6 ARIMA111211 0.00000155   3133. -6255. -6255.
## 7 ARIMA111112 0.00000155   3133. -6255. -6255.
## 8 ARIMA211211 0.00000155   3134. -6253. -6253.
## 9 ARIMA112112 0.00000155   3133. -6253. -6253.
```

```
ari.fit3 %>% select(ARIMA011011) %>% gg_tsresiduals(lag = 36)
```




```
augment(ari.fit3) %>%
  features(.innov, ljung_box, lag=200, dof=4)
```

```
## # A tibble: 9 × 3
##   .model      lb_stat lb_pvalue
##   <chr>         <dbl>     <dbl>
## 1 ARIMA011011    132.      1.00
## 2 ARIMA013011    131.      1.00
## 3 ARIMA111111    130.      1.00
```

```
## 4 ARIMA111112    130.      1.00
## 5 ARIMA111211    131.      1.00
## 6 ARIMA112111    129.      1.00
## 7 ARIMA112112    131.      1.00
## 8 ARIMA211211    131.      1.00
## 9 auto           132.      1.00
```

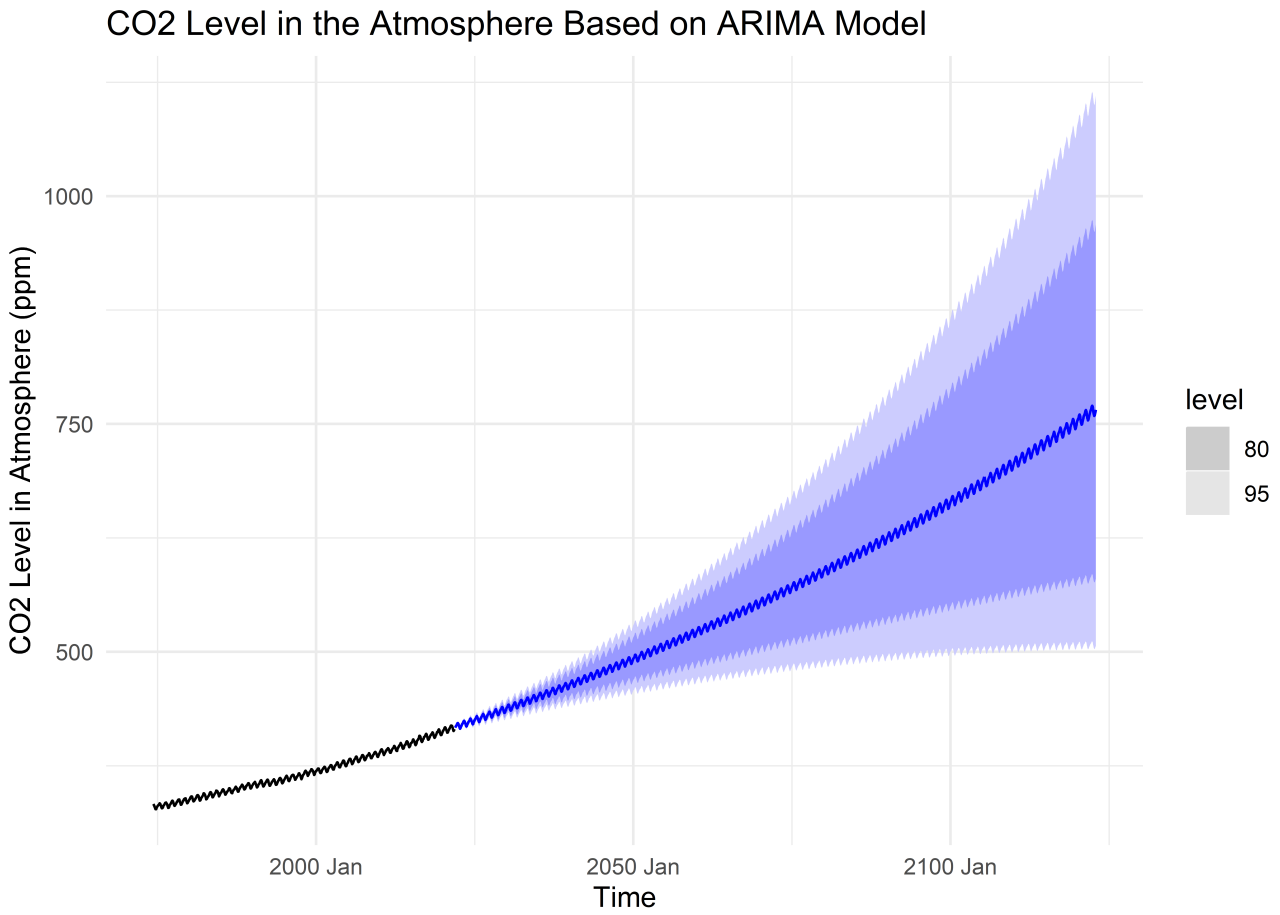
```
mul.quad.sea3 <- co2.log.nsa.tr %>%
  model(trend_model = TSLM(log_value ~ trend() + I(trend()^2) + season()))

future.df3 <- new_data(co2.log.nsa.tr, n = 24)
lm.fx.te2 <- mul.quad.sea3 %>% forecast(new_data = future.df3)

#Out of sample test
bind_rows(
  ari.fit3 %>% select(ARIMA011011) %>% forecast(h = 24) %>% fabletools::accuracy(co2.
  ari.fit3 %>% select(ARIMA111111) %>% forecast(h = 24) %>% fabletools::accuracy(co2.
  ari.fit3 %>% select(ARIMA111211) %>% forecast(h = 24) %>% fabletools::accuracy(co2.
  ari.fit3 %>% select(ARIMA211211) %>% forecast(h = 24) %>% fabletools::accuracy(co2.
  ari.fit3 %>% select(ARIMA112112) %>% forecast(h = 24) %>% fabletools::accuracy(co2.
  ari.fit3 %>% select(ARIMA112111) %>% forecast(h = 24) %>% fabletools::accuracy(co2.
  ari.fit3 %>% select(ARIMA111112) %>% forecast(h = 24) %>% fabletools::accuracy(co2.
  ari.fit3 %>% select(ARIMA013011) %>% forecast(h = 24) %>% fabletools::accuracy(co2.
  lm.fx.te2 %>% fabletools::accuracy(co2.log.nsa.te)
)
```

```
## # A tibble: 9 × 10
##   .model      .type         ME    RMSE     MAE      MPE   MAPE  MASE RMSSE  ACF1
##   <chr>       <chr>      <dbl>   <dbl>   <dbl>    <dbl>  <dbl> <dbl> <dbl> <dbl>
## 1 ARIMA011011 Test  -0.000871  0.00148 0.00111 -1.44e-2 0.0184   NaN   NaN 0.418
## 2 ARIMA111111 Test  -0.000797  0.00144 0.00108 -1.32e-2 0.0178   NaN   NaN 0.425
## 3 ARIMA111211 Test  -0.000795  0.00144 0.00108 -1.32e-2 0.0178   NaN   NaN 0.429
## 4 ARIMA211211 Test  -0.000796  0.00144 0.00108 -1.32e-2 0.0178   NaN   NaN 0.429
## 5 ARIMA112112 Test  -0.000796  0.00144 0.00108 -1.32e-2 0.0178   NaN   NaN 0.428
## 6 ARIMA112111 Test  -0.000813  0.00145 0.00108 -1.35e-2 0.0179   NaN   NaN 0.416
## 7 ARIMA111112 Test  -0.000799  0.00144 0.00108 -1.32e-2 0.0178   NaN   NaN 0.424
## 8 ARIMA013011 Test  -0.000853  0.00147 0.00110 -1.41e-2 0.0182   NaN   NaN 0.418
## 9 trend_model Test  -0.0000166 0.00137 0.00116 -2.66e-4 0.0192   NaN   NaN 0.443
```

Then, the team developed the models using the non-seasonally adjusted data. The results show that all models produced a similar level of AIC score and out-of-sample MAPE score. We selected the ARIMA model with a seasonal and non-seasonal ma term with one seasonal and non-seasonal differencing since this is a simpler model that performs

reasonably well. Interestingly, we also found that the linear model with the updated data does not perform as well as the ARIMA model.

Future Predictions



According to the selected model (ARIMA model with one seasonal and non-seasonal difference term, one seasonal and non-seasonal moving average term), the CO2 level is expected to reach 420 ppm first time in 2022 Apr and last time in 2024 Sep. The 90% CI of the model prediction suggests that we may see the CO2 level reach 420 ppm as early as in 2022 Feb and may see it again as late as in 2026 Sep.

According to the model, the CO2 level is expected to reach 500 ppm. first time in 2052 Apr and last time in 2054 Sep. The 90% CI of the model prediction suggests that we may see the CO2 level reach 500 ppm as early as in 2044 Apr and may see it again as late as in 2082 Oct.

Conclusion

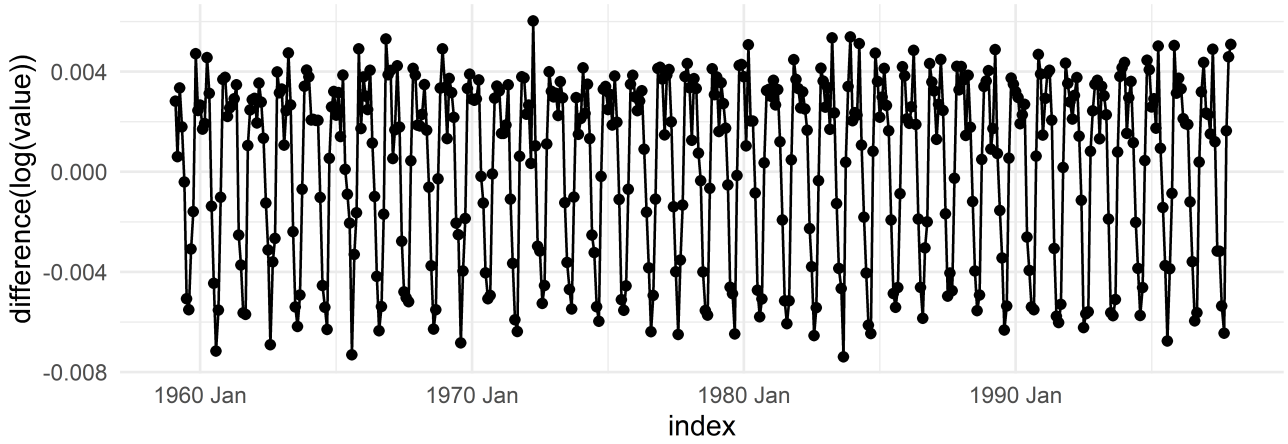
Our journey from 1997's historical insights to the contemporary CO2 dynamics at Mauna Loa uncovered both continuities and shifts in atmospheric CO2. Armed with updated data, we aimed to reassess prior forecasts and discern any systematic changes in CO2 growth patterns.

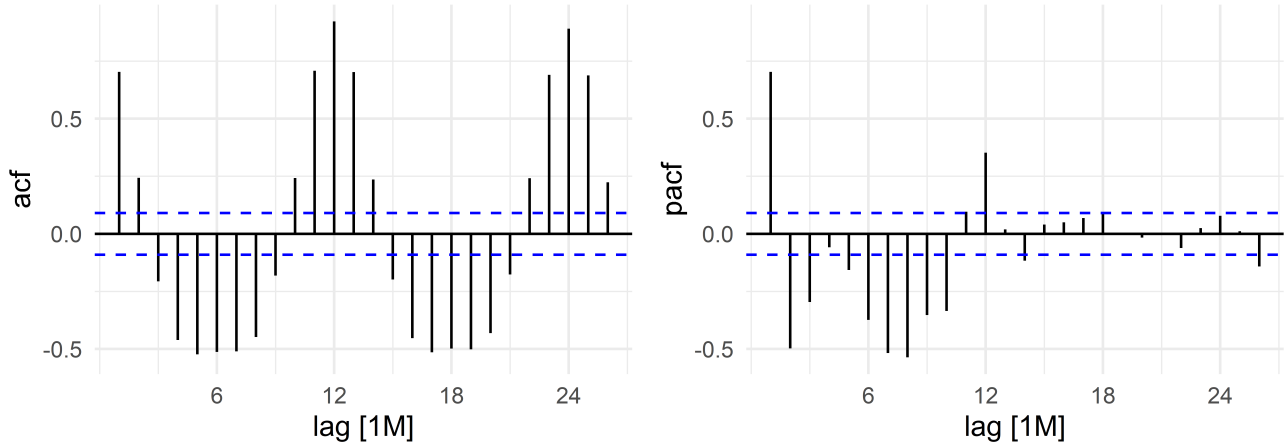
Our prior, 1997 models demonstrated divergence from modern conditions, revealing overforecasting from 2000 to 2020. In the realm of ARIMA models, we observed CO2 levels were near the upper bounds of the 80% confidence interval, hinting at an increase in the underlying contributor sources to atmospheric CO2. The ARIMA forecast consistently projected lower values yet, maintained a proximity to its upper confidence interval.

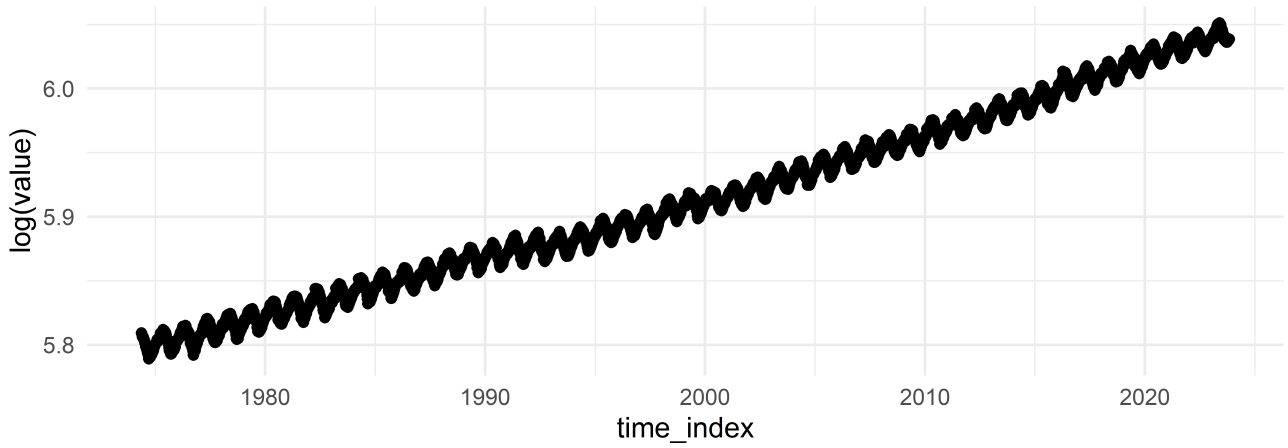
With insights from 1997 models and new data, our gaze extends to the future. The interplay of forecasts suggests that the ARIMA model, with the addition of seasonal and non-seasonal terms, offers a more accurate prediction of atmospheric CO2. The Ljungbox test underscores the model's efficacy with non-seasonally adjusted data. Looking ahead, our selected ARIMA model predicts a first encounter with a concentration of 420 ppm as early as 2022 Apr and a lingering farewell as late as 2024 Sep. The anticipation of reaching 500 ppm, a milestone with extreme environmental implications, is also predicted for the future. Here, we acknowledge the inherent uncertainty accompanying long-term projections and hope that un-incorporated factors, such as the adoption of new technologies and carbon mitigation techniques will render our forecasts as worst-case over-predictions.

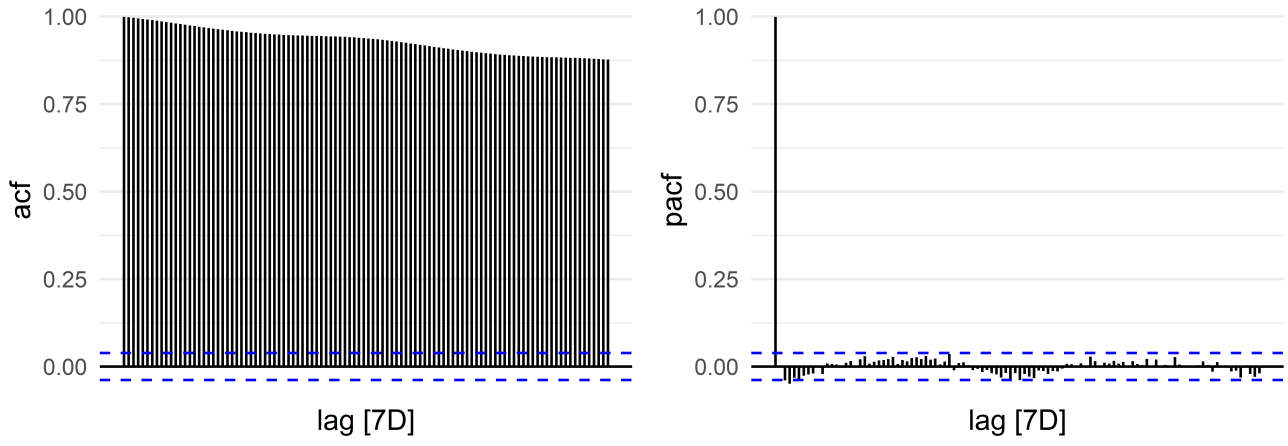
Appendix

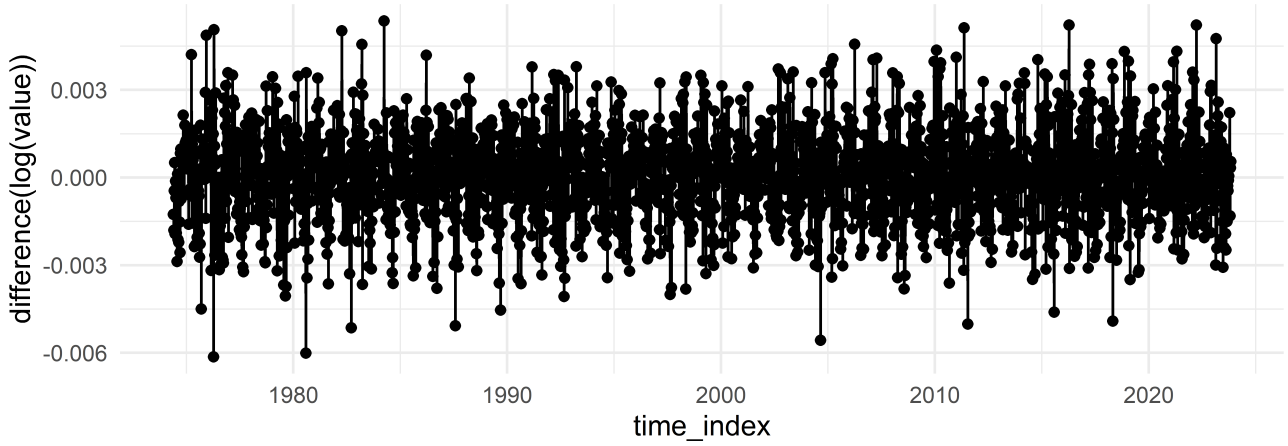
Iterative Differencing Plots

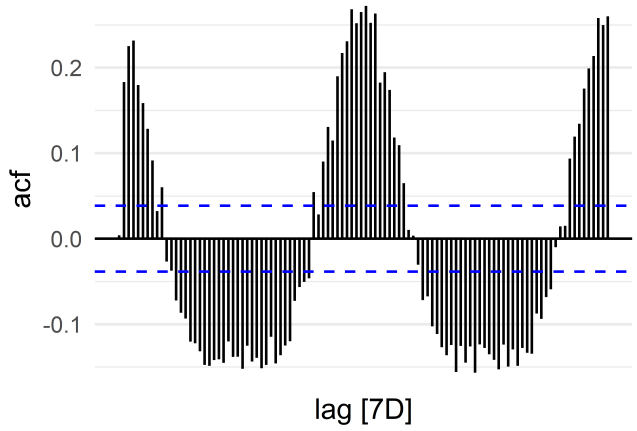

difference(log(value))
0.004
0.000
-0.004
-0.008
1960 Jan
1970 Jan
1980 Jan
1990 Jan
index

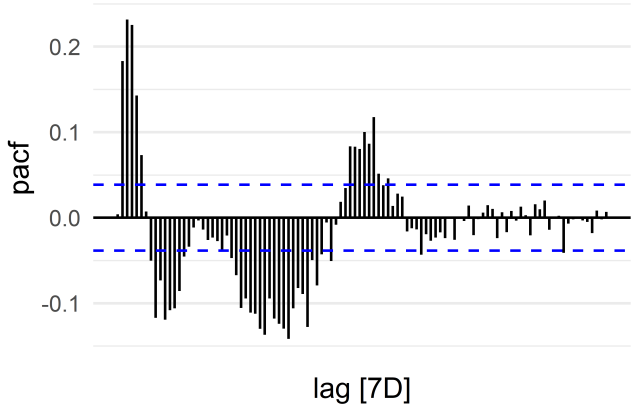

acf
0.5
0.0
-0.5
6
12
18
24
lag [1M]
pacf
0.5
0.0
-0.5
6
12
18
24
lag [1M]

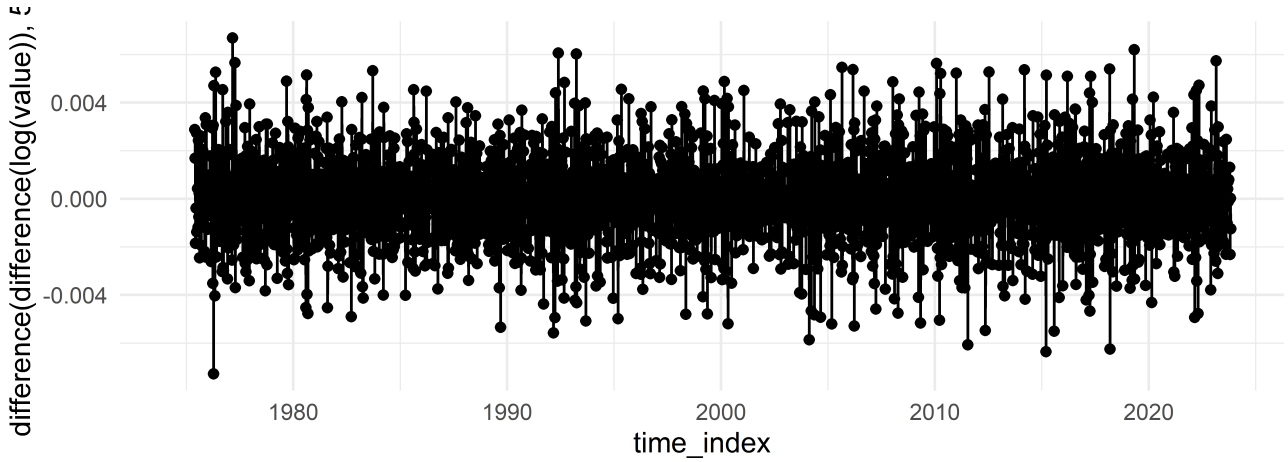

log(value)
6.0
5.9
5.8
1980
1990
2000
2010
2020
time_index

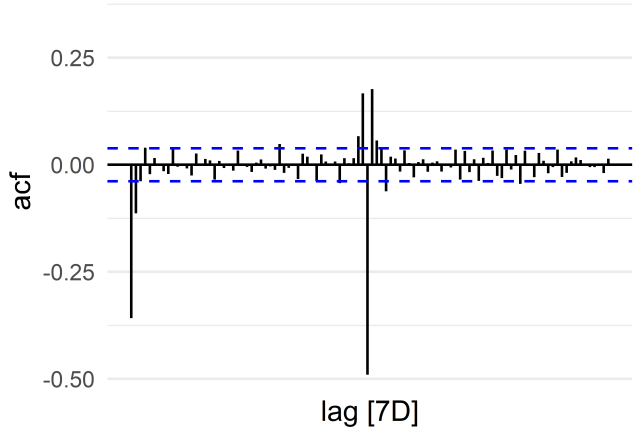

acf
1.00
0.75
0.50
0.25
0.00
lag [7D]
pacf
1.00
0.75
0.50
0.25
0.00
lag [7D]

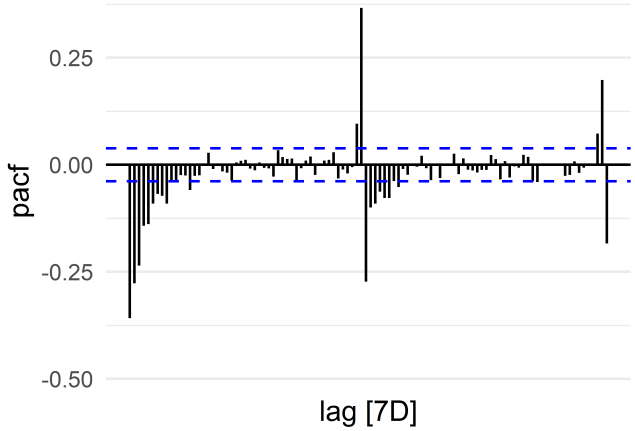

difference(log(value))
0.003
0.000
-0.003
-0.006
1980
1990
2000
2010
2020
time_index


acf
0.2
0.1
0.0
-0.1
lag [7D]


pacf
0.2
0.1
0.0
-0.1
lag [7D]


difference(difference(log(value)),
0.004
0.000
-0.004
1980
1990
2000
2010
2020
time_index


acf
0.25
0.00
-0.25
-0.50
lag [7D]


pacf
0.25
0.00
-0.25
-0.50
lag [7D]