Comparing Auto-ARIMA to NNAR

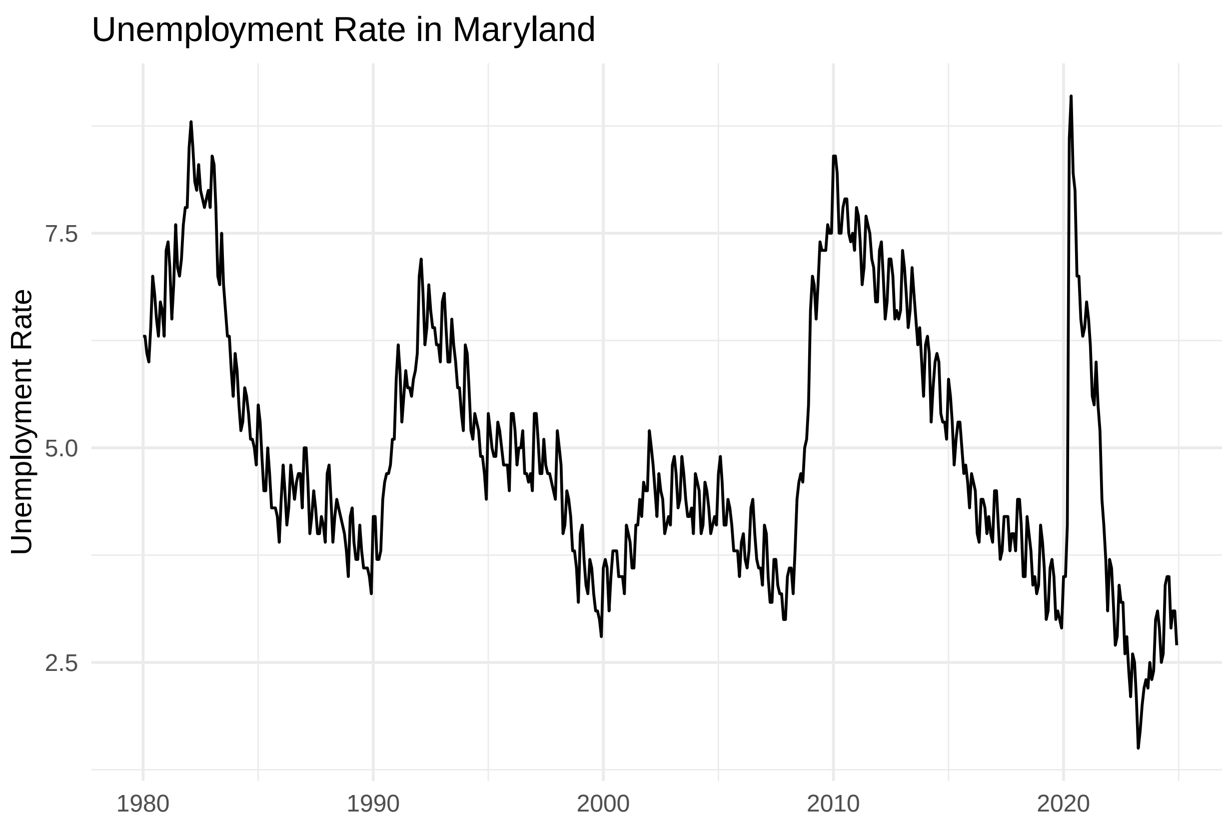
Luke Harold

2025-09-10

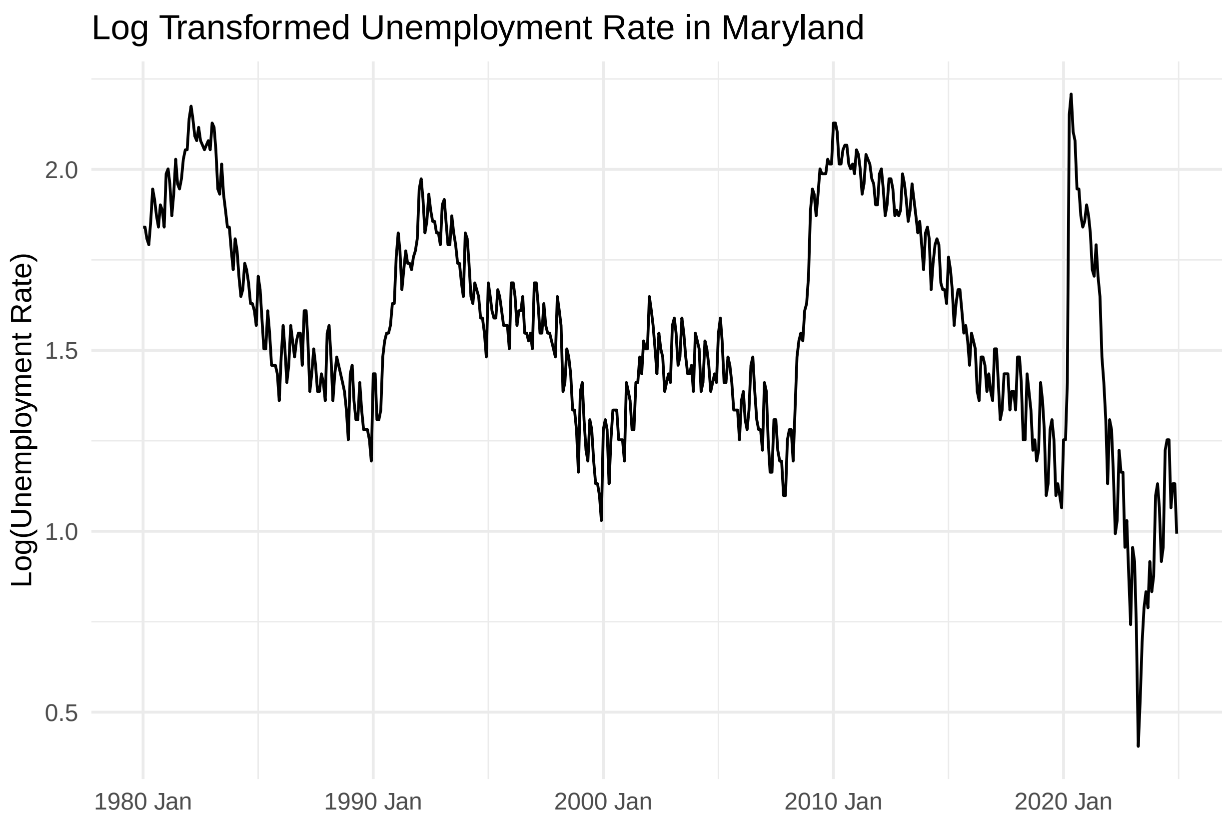
Introduction

The goal of this assignment was to build a model to forecast the unemployment rate for an assigned U.S. state, in this case, Maryland. The training window ranged from 1980 to December 2012, with the goal of producing a recursive forecast from 2013 onward. There are several important considerations in constructing an accurate forecasting model. This paper focuses on comparing three traditional time series models using recursive forecasting, followed by an attempt to use a Neural Network Autoregression (NNAR) model with an adaptive training window to see whether it can outperform the classic models. The goal was to determine whether the NNAR model can better account for structural features of the time series, such as cyclicality and time-varying seasonality.

#loading data

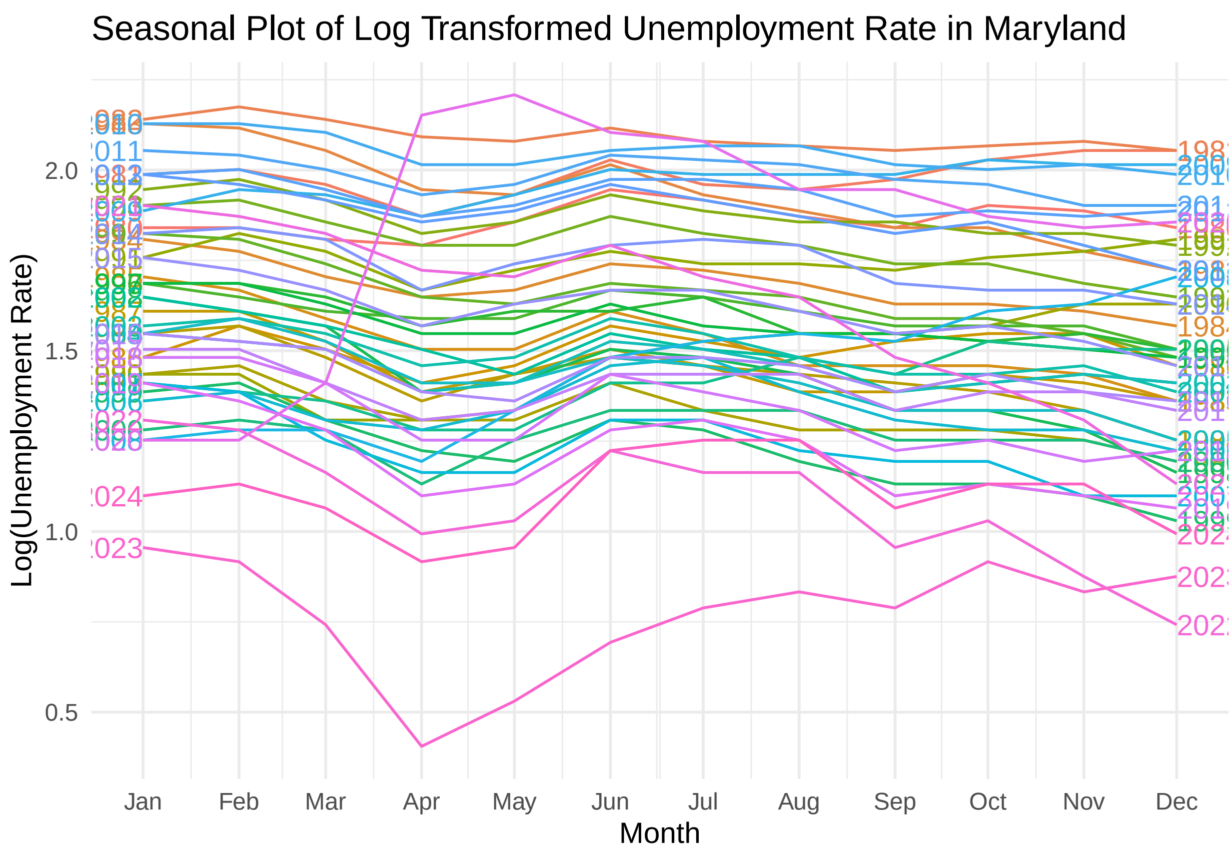


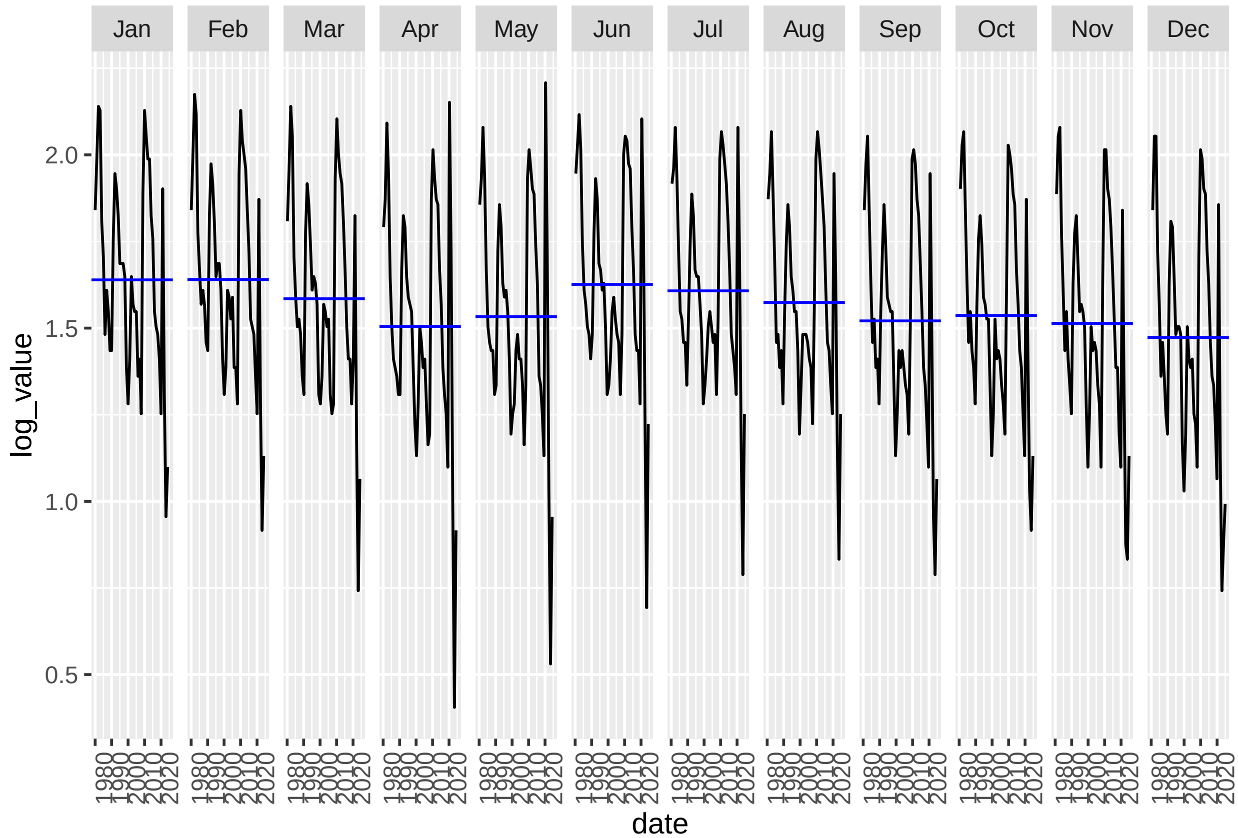
Visualy the data exhibits a few key features including cylclicallity, seasnality and a downward trend. In order to combat this we can apply a log transformation to stabilize the variance and then decompose the series to better understand its components. By applying a log transformation, we can stabilize the variance and make the series more stationary, which is often a prerequisite for many time series forecasting methods. As we will be using Auto Arima to choose a model to compare with the Neural Network Autoregression (NNAR) model, we won't pre difference the data.

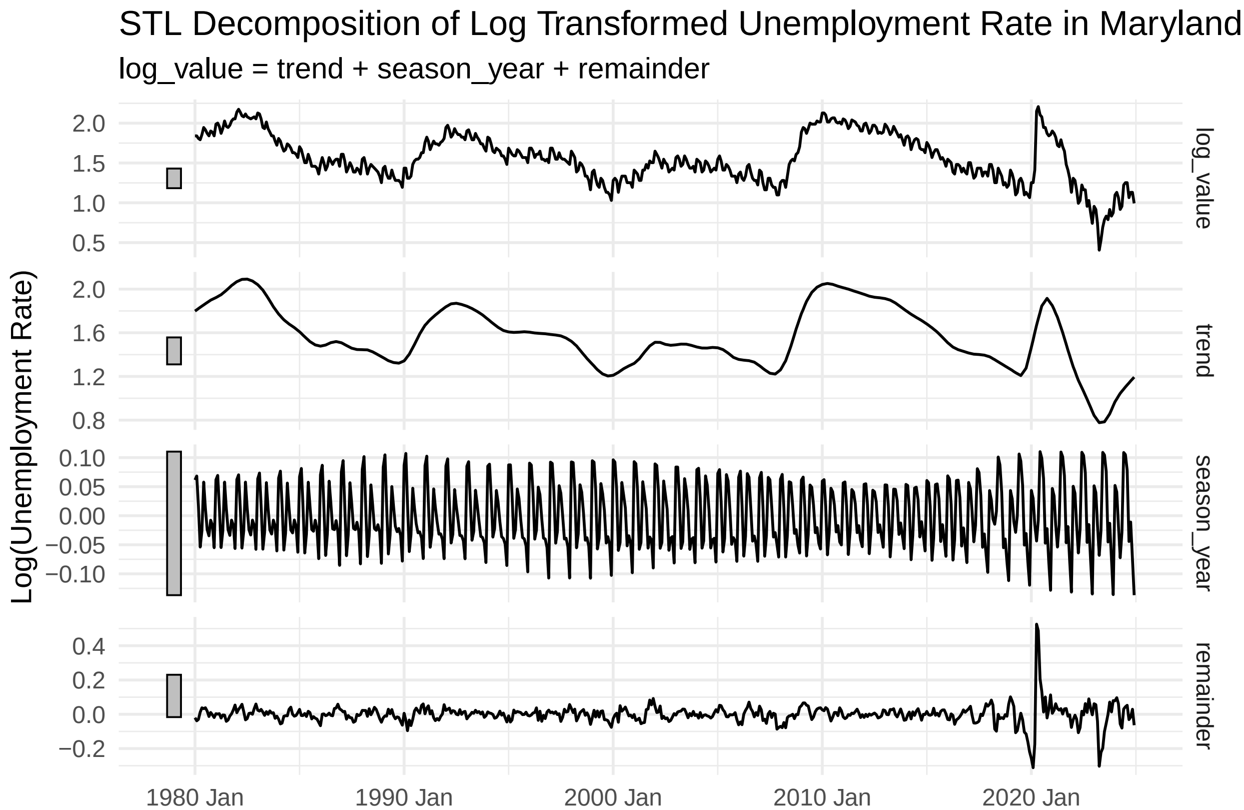


investigating the changing seasonality and structural breaks: Given that the there is seasonality in the data, it is interested to visualize whether this component remains constant over time or if these trends change. With an individual state's unemployment rate it is possible that between 1980-2024 the industry composition of the state has changed, which could lead to changes in seasonal patterns. To investigate this we can use a seasonal subseries plot to visualize the seasonality over time.

Including the covid period may be more indicative of when restrictions were put in place and the unemployment rate spiked, but it is also a structural break in the data that may not be representative of future trends. If we deconstruct the unemployment rate into 20 year periods we see that the seasonal patterns appear to change slightly.


Seasonal Plot of Log Transformed Unemployment Rate in Maryland
Log(Unemployment Rate)
2.0
1.5
1.0
0.5
Jan
Feb
Mar
Apr
May
Jun
Jul
Aug
Sep
Oct
Nov
Dec
Month


Jan
Feb
Mar
Apr
May
Jun
Jul
Aug
Sep
Oct
Nov
Dec
log_value
2.0
1.5
1.0
0.5
1980
1990
2000
2010
2020
date



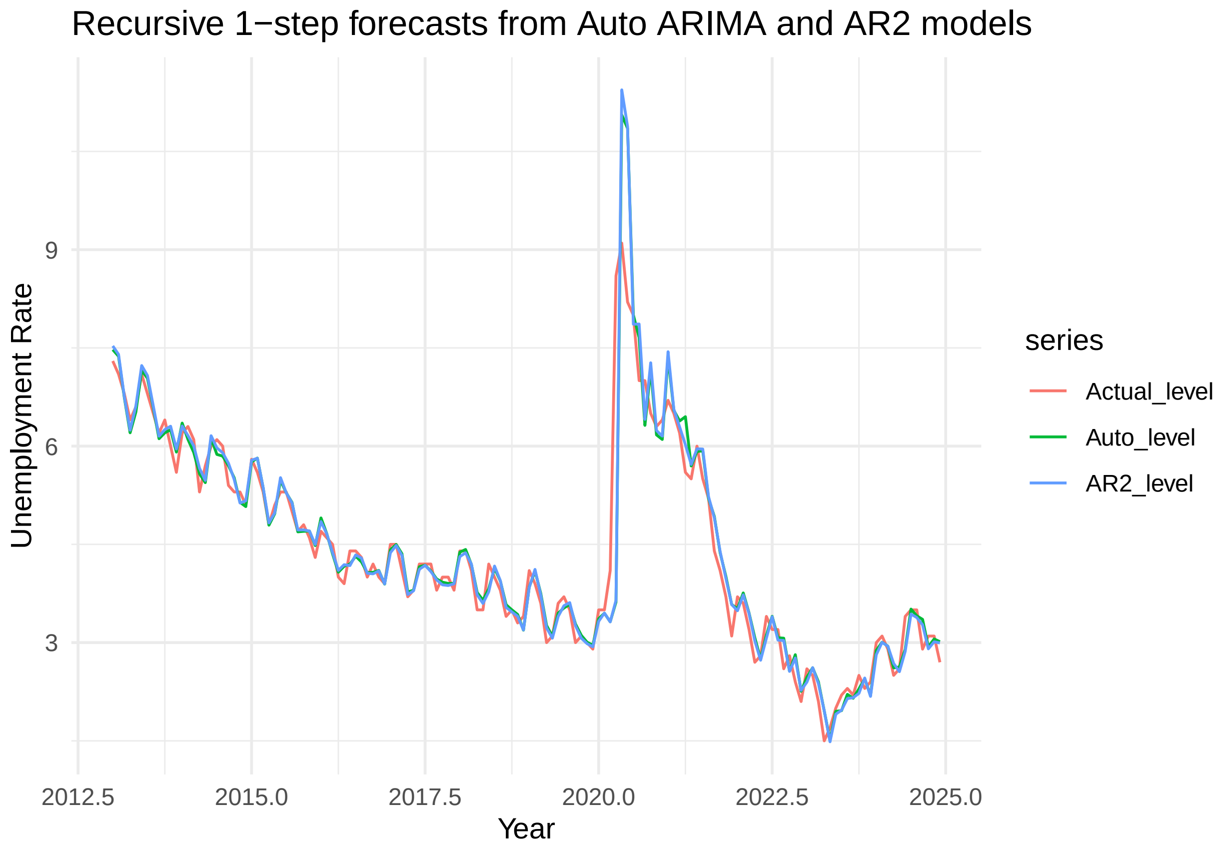

```
## # A mable: 2 x 2
## # Key:     Model name [2]
##   `Model name`                    Orders
##   <chr>                          <model>
## 1 AR2          <ARIMA(2,1,0)(0,1,1)[12]>
## 2 auto                      <NULL model>
```

```
## # A tibble: 1 x 6
##   .model  sigma2 log_lik    AIC   AICc    BIC
##   <chr>    <dbl>   <dbl>  <dbl>  <dbl>  <dbl>
## 1 AR2    0.00135    717. -1426. -1426. -1410.
```

#Using auto arima and a manual ARIMA(2,1,0) to compare with NNAR we determine that the model selected by auto arima to have residuals that are white noise whereas the residuals from the AR2 are not. Evidence from the ljung box test shows that the p-value for the auto arima model is above 0.05, indicating that we fail to reject the null hypothesis that the residuals are white noise. In contrast, the AR2 model has a p-value below 0.05, suggesting that its residuals are not white noise and that there may be some autocorrelation present.

```
## Time difference of 2.224462 mins
```

```
##       rmse_AR2 rmse_auto
## [1,] 0.5629565 0.5534937
```



Comparing the two models we see that the auto arima model has a lower rmse than the AR2 model, but visually we can see that both struggle greatly to capture the spike in unemployment during the covid period. Is there potentially a model that can better capture these structural breaks and changing seasonal patterns? Neural Network Auto-regression (NNAR) is a forecasting method that uses a feed-forward neural network to model the relationship between past values of a time series and its future values. It is used for its ability to capture complex patterns in the data and is not constrained by the assumptions of traditional time series models.

Neural Network Auto-regression uses several parameters including the number of lagged inputs(p), the number of seasonal lagged inputs (P), the number of nodes in the hidden layer (size), and the number of times to repeat the fitting process with different random starting weights (repeats). These parameters can be changed to see which combination fits the data best. Here we train a NNAR on the training window to find the best structure, which we will then use in the recursive forecast allowing the weights to change but keeping the 5 lags, 1 seasonal lag and 4 hidden nodes.

```
## Series: train0
## Model:  NNAR(5,1,4)[12]
## Call:   forecast::nnetar(y = train0)
##
## Average of 20 networks, each of which is
## a 6-4-1 network with 33 weights
## options were - linear output units
##
```

```
## sigma^2 estimated as 0.002915
```

Here we see that it outperforms the arima models from earlier.

```
##       rmse_AR2 rmse_auto rmse_nnar
## [1,] 0.5629565 0.5534937 0.5442734
```

We also will fit models allowing the NNAR to change for each forecast of the data and with a dynamic window. The dynamic window uses a NNAR with the same structure as above but allows the training window to grow with each step, starting with a minimum of 3 years and increasing by 1 year at each step. This allows the weights to change and the window to change based on the length of the window that produced the most accurate forecast in the previous step. For t+1, the window that produced the smallest error for t is used to produce the forecast for t+1 and so forth. The idea here is that in uncertain times like covid, using lots of historical data may not actually be helpful in producing accurate forecasts.

```
## Time difference of 2.743263 mins
```

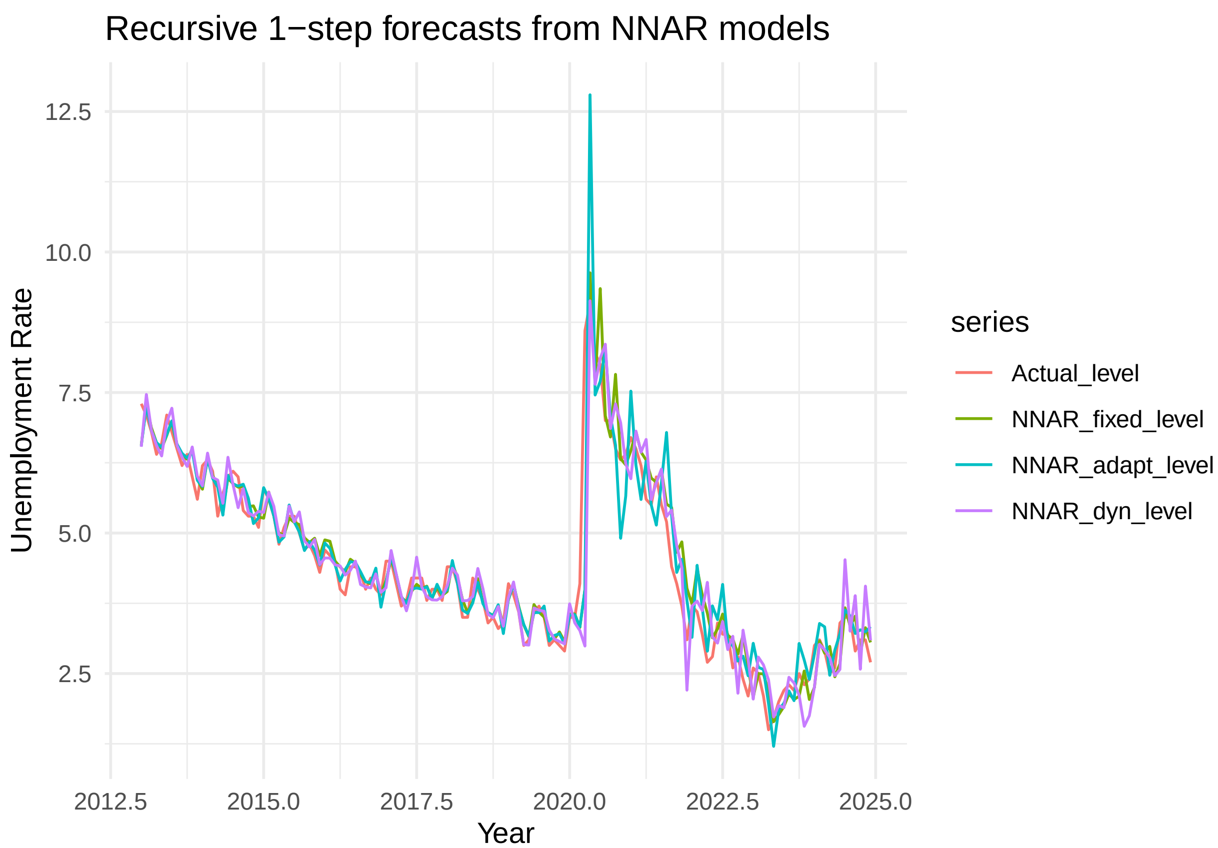
```
## [1] 0.6316459
```

```
##       rmse_AR2 rmse_auto rmse_nnar rmse_nnar_adaptive
## [1,] 0.5629565 0.5534937 0.5442734          0.6316459
```

```
## Time difference of 9.045441 mins
```

```
## [1] 0.6357008
```

```
##       rmse_AR2 rmse_auto rmse_nnar rmse_nnar_adaptive rmse_nnar_dyn
## [1,] 0.5629565 0.5534937 0.5442734          0.6316459     0.6357008
```



We see that only the NNAR that was fit on the entire training window outperforms the auto arima model and AR2 model across the entire series.

Table 1: RMSE for all models

rmse_AR2	rmse_auto	rmse_nnar	rmse_nnar_adaptive	rmse_nnar_dyn
0.5629565	0.5534937	0.5442734	0.6316459	0.6357008

How do the models compare specifically during the covid period? From the results we see that the NNAR model that was fit on the training data that keeps the same structure throughout performs the best.

```
## # A tibble: 5 x 2
##   model        RMSE
##   <chr>       <dbl>
## 1 NNAR_fixed  1.42
## 2 NNAR_adapt  1.77
## 3 NNAR_dyn    1.67
## 4 autoARIMA   1.71
## 5 AR2         1.76
```

From this analysis we see that NNAR can be used to effectively forecast univariate time series data and the fixed model outperformed the arima models along with the adaptive and dynamic models.