Modeling SNAP Benefit Recipients in California

Econ 144 - Project 3

Jacob Titcomb

Spring 2023

Contents

I. Introduction

Our project attempts to model and forecast SNAP benefit recipients for California. SNAP, or Supplemental Nutrition Assistance Program, is a federally mandated social welfare program that aims to provide food and resources to low-income individuals and families across the nation. The program was created in 1939 and it is sometimes known by its former name of "Food Stamps;" in 1977 it was retrofitted and renamed to the "Supplemental Nutrition Assistance Program" (*Food and Nutrition Service*). According to the USDA Food and Nutritional Service, in 2017, one sixth of all children in the U.S. lived in households with SNAP benefits, and in 2018, the SNAP program spanned 20 million households and 40 million individuals in the United States. Understanding SNAP and its number of recipients over time provides a glimpse into the nature of poverty in the United States and insight into the low-wage labor market.

In California, SNAP has its own name of CalFresh. It is the largest food program in California; though the program is state-supervised, it is operated on the county-level (*Department of Social Services*). The most basic eligibility requirement in California is that an individual's maximum gross income does not exceed 200% the federal poverty level. That restriction varies depending on household income, homelessness status, past drug offences, immigration status, and other factors (*Department of Social Services*). Like other SNAP programs around the US, the CalFresh program was heavily affected by the COVID-19 pandemic and the economic instability that it caused.

The data set we used was sourced from the Federal Reserve Bank of St. Louis (*FRED*). In order to make the data more stationary, we took the first difference; thus we are studying the *changes* in California SNAP recipients. To make the scale of the data more interpretable, we also divided by 1000, so all observations are in terms of thousands of persons. The frequency of the data is monthly and it was not seasonally adjusted. Lastly, the window of the data spans from February 1981 to June 2021.

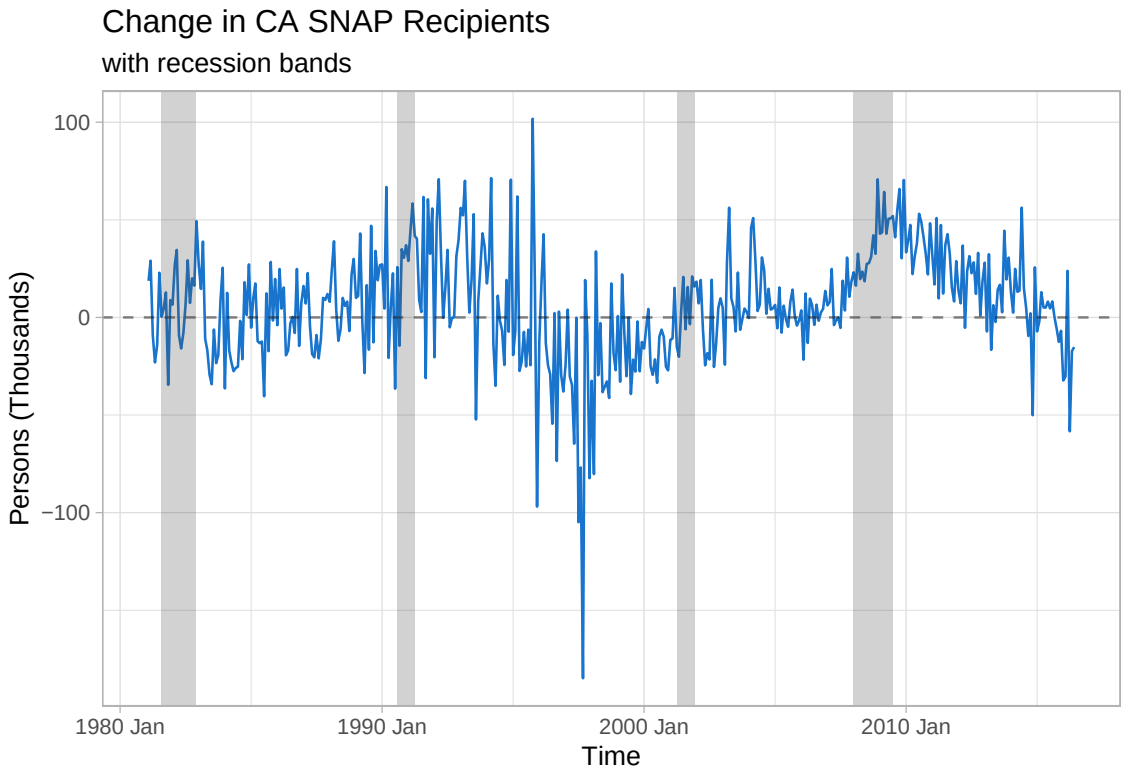
For our VAR model, we chose to include changes in SNAP recipients for the state of Washington as an exogenous variable. The window of time for Washington is the same and we made the same transformations to the Washington data set as we did California.

In order to evaluate the performance of our models, we performed a train-test split of the data. The test data set is the last 5 years: July of 2016 to June 2021. We will evaluate our models using mean absolute error (MAE) and root mean squared error (RMSE). We would have also used mean absolute percent error (MAPE), but some of our data points are 0, so we cannot calculate MAPE. We also use mean error (ME) to evaluate whether our forecasts over-/under-estimate the true values.

II. Results

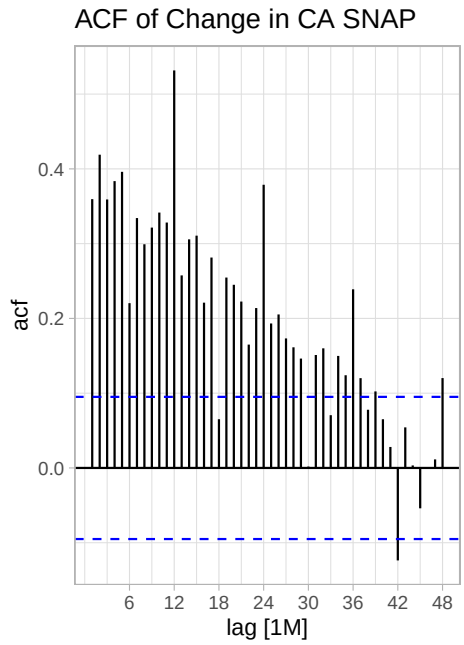
(a) Time Series Plots

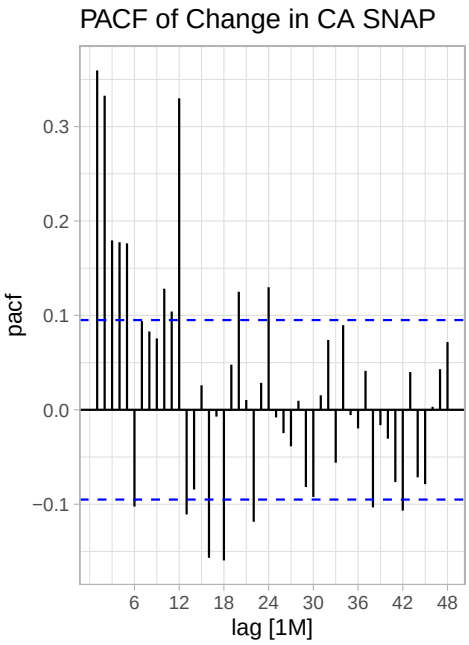
In the time plot below, we see a generally flat trend to the data with non-constant variability. There are very prominent cycles with strong persistence. A seasonal component is likely present, but it is difficult to tell at this scale, as it is not obvious. In terms of an ARMA framework, the time series tends more towards AR behavior, with slower mean reversion and some periods of strong persistence. The data appears to be stationary with a mean close to 0. We calculated the mean to be 6.329541 and the standard deviation to be 29.56348, both in thousands of persons.



ACF and PACF

As far as the ACF and PACF for the data, we see prominent spikes at lags that are multiples of 12 in both plots. The ACF plot shows those lags as well – as all the other lags too – decaying. Whereas the PACF shows a significant lag at 12 and *maybe* 24. Thus there is strong seasonality which likely follows a seasonal AR(1) process ($s = 12$). For the non-seasonal lags, as mentioned before, the ACF has the lags decaying, while the PACF as some significant lags, but after lag 18 the rest are not significant. This is consistent with an AR process, supporting our assertion from earlier. For an ARMA model of the non-seasonal component, we would initially propose an AR(5) or AR(18) model. Regardless, the messy behavior in the PACF indicates that further differencing in an ARIMA model will likely be necessary.

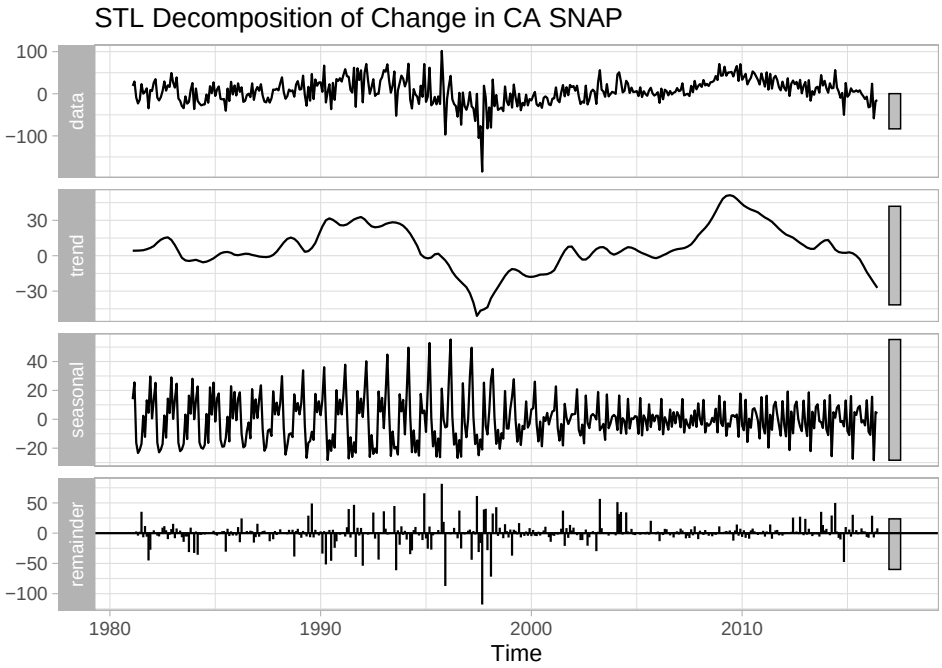

ACF of Change in CA SNAP
acf
0.4
0.2
0.0
6
12
18
24
30
36
42
48
lag [1M]


PACF of Change in CA SNAP
pacf
0.3
0.2
0.1
0.0
−0.1
6
12
18
24
30
36
42
48
lag [1M]

(b) STL + ARMA Model

Trend, Seasonality, and Cycles

Below, we constructed an STL decomposition of the time series with changing seasonal factors. The trend component follows the series fairly closely, especially in the periods with low levels of noise. For the seasonal component, we selected a seasonal window of 4 years (48 months). We see a slightly decreasing amplitude of the seasonal component, but the amplitude is fairly large throughout, considering the scale of the trend. Lastly, we note that the scale of the random component is large compared to the trend and seasonal components, so that could be evidence of cycles. The remainder component shows few patterns if any, indicating that the cycles potentially follow either a white noise or MA process.



Model Construction

We propose the following model:

$$Y_t = T_t^{Loess} + \sum_{i=1}^{12} \delta_i M_{i,t} + R_t \qquad \text{Trend and Seasonality}$$

$$R_t = \theta \varepsilon_{t-1} + \varepsilon_t \qquad \text{Cycles}$$

The first term (T_t^{Loess}) is the Loess fitted model for trend, found using the STL decomposition above. The second component of the model captures seasonality: the general monthly behavior of the series, with a window of 4 years. Lastly, we selected an MA(1) model for the cycles, represented by the equation with R_t. We chose the order of 1 using the `auto.arima()` function, which we also used to determine the estimate for θ.

We are unable to extract the explicit Loess parameters for the trend component, but we can extract the seasonality and cycle components. The averaged seasonality component and the cycle component are summarized in the following two tables:

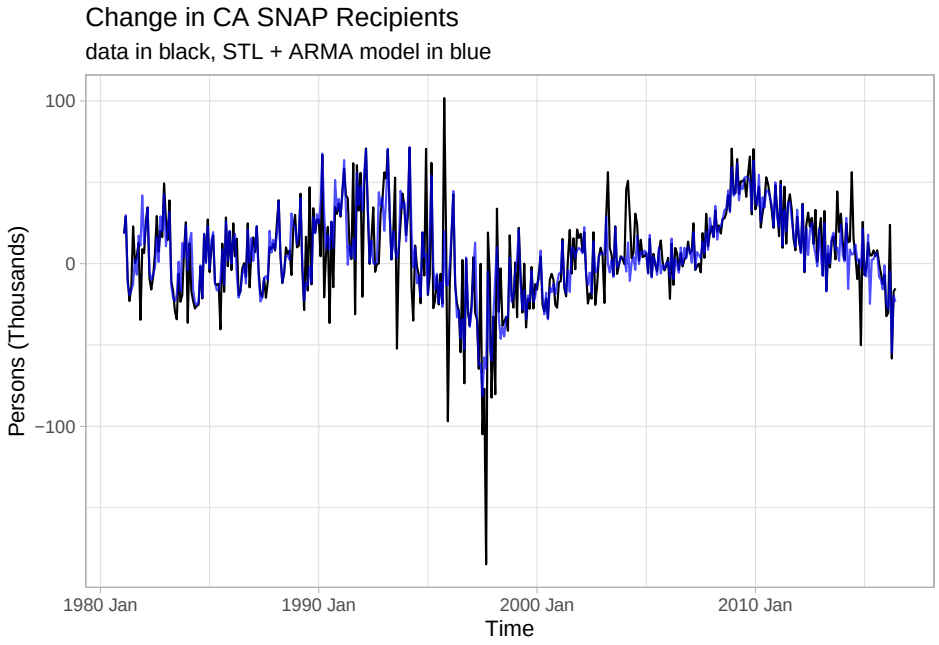
Table 1: CA Mean Seasonal Component

Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
-1	6	26	-9	-12	-9	-12	0	-10	10	-1	10

Table 2: CA Cycle Component

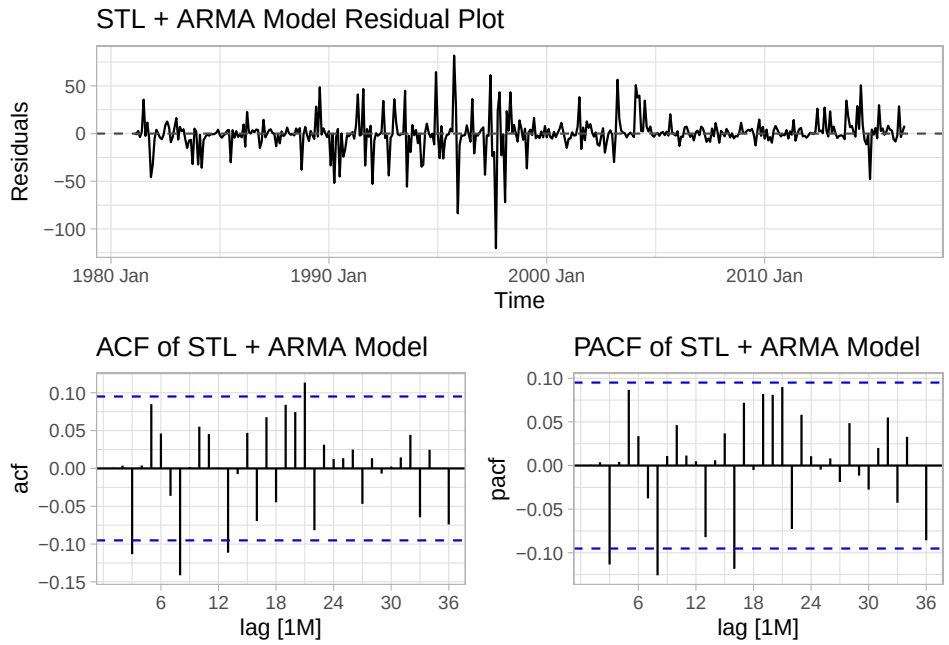
	MA(1)
Coefficients	-0.1247

To illustrate the fit of the model, the following graph shows the original time series (black curve) and the model (blue curve).

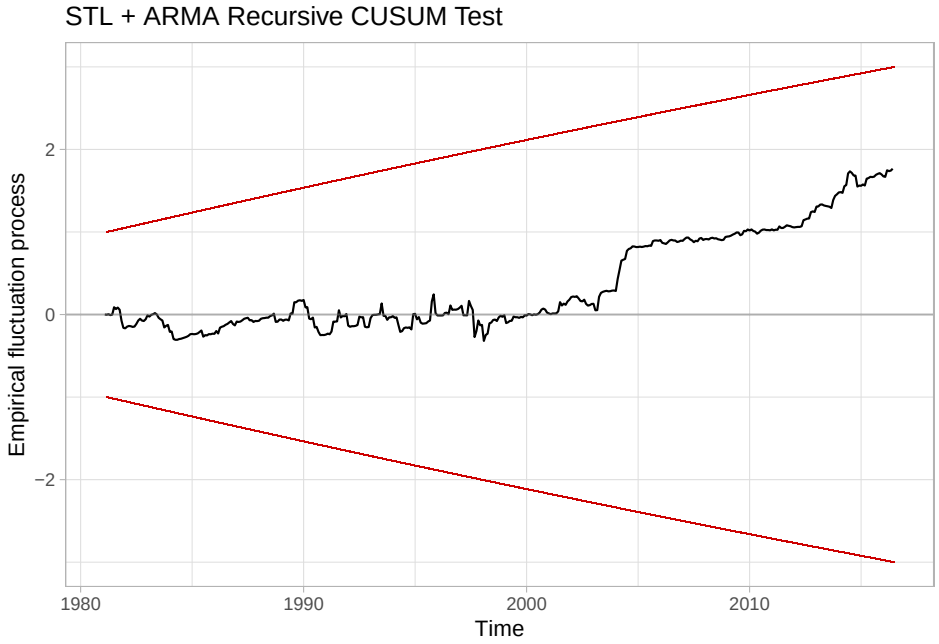


Assessing Model Validity

The time plot of the residuals still shows some structure, indicating that our STL decomposition did not fully capture the trend or our ARMA model did not fully capture the cycles, with the latter being more likely. The ACF and PACF show that much, if not all, of the serial correlation has been captured by the model. In particular, we observe very few significant spikes, and those that are significant have small magnitudes. We will further analyze serial correlation when covering model diagnostic statistics.



Lastly, we look to the cumulative sum plot to identify any structural breaks. The CUSUM plot shows no significant divergence from 0, implying that there are no structural breaks in our model.



Model Diagnostic Statistics

```
## --- Training Dataset ---
## MAE:  9.598892
## RMSE: 17.503
```

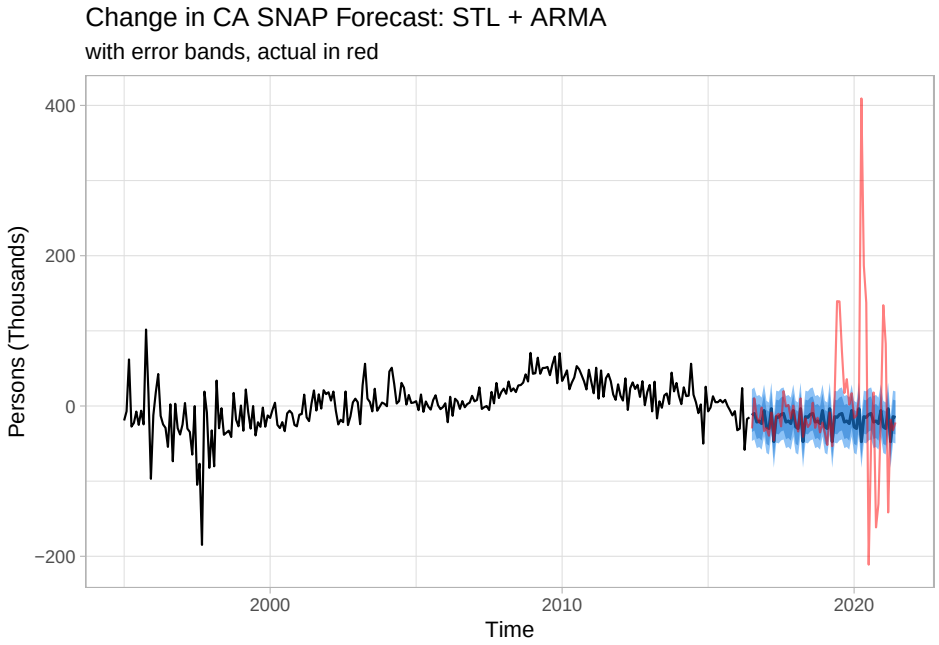
```
## ME:   0.06617322
```

```
##
##  Box-Ljung test
##
## data:  resids
## X-squared = 47.537, df = 24, p-value = 0.002879
```

The model – which included an STL decomposition with an ARMA component for cycles – has a high MAE and RMSE, at 9.598892 and 17.503, respectively. Compared to the scale of the time series which has a standard deviation of approximately 30, a mean absolute error of 9.598892 is quite large. Along with the large RMSE, these values are too high for a suitable forecast, thus we do not expect this to perform well on the testing data. From the mean error (ME), we also can see that our model slightly underestimates the true values.

A Ljung-Box test on 24 lags resulted in a p-value of 0.002879, so at 0.05 significance there is evidence of serial correlation in the residuals. This test further shows that the model is ineffective: an ideal model should completely wipe out all serial correlation. Therefore we can infer that this model fails to capture aspects of the data, such as trend or (more likely) cycles.

Forecasting



In the plot above, we have a 5-year forecast for the time series with the true data overlaid in red. Up until around 2019, our model does fairly well at capturing the behavior. However, the extreme behavior that starts around the end of 2019 results in our estimates being drastically off. This shortcoming is evident in the goodness-of-fit measures calculated for the testing set: the MAE and RMSE are very high, 46.29502 and 88.74208, respectively. And the ME shows that we underestimated the true behavior by 18.38043 on average.

```
## --- Testing Dataset ---
## MAE:  46.29502
## RMSE: 88.74208
## ME:   18.38043
```

(c) ARIMA Model

Model Construction

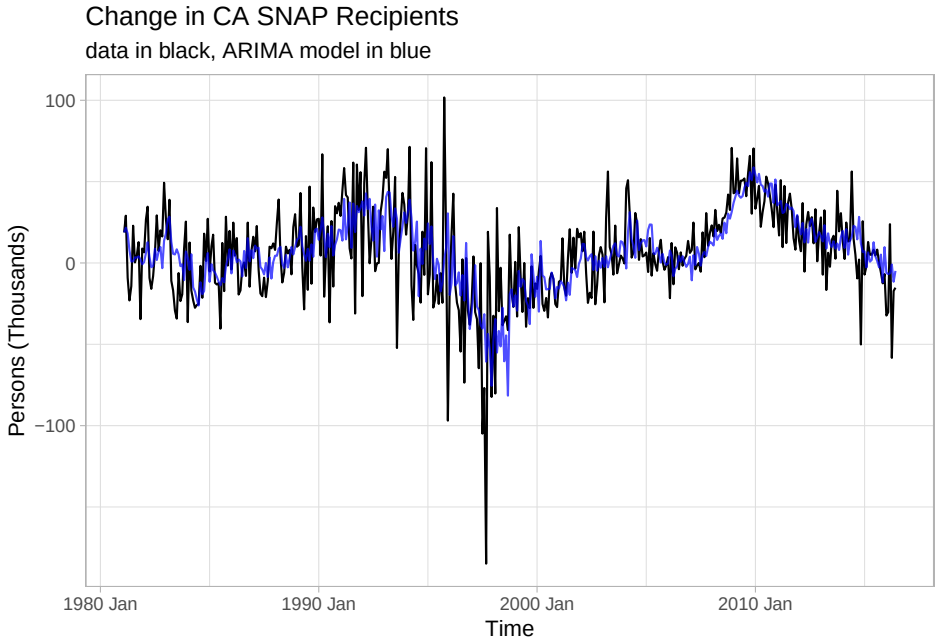
We used the `auto.arima()` function in R to estimate an ARIMA model. The summary of the model is in the code output below. The algorithm determined the optimized model to be an ARIMA$(0, 1, 1)$ model with a seasonal MA(2) component ($s = 12$), taking the following form:

$$\text{Trend and Cycles:} \quad (1 - L)y_t = (1 - 0.8451L)\ \varepsilon_t$$

$$\text{Seasonal:} \quad y_t = (1 + 0.2868L^{12} + 0.2429L^{24})\varepsilon_t$$

```
## Series: CA_ts
## ARIMA(0,1,1)(0,0,2)[12]
##
## Coefficients:
##           ma1    sma1    sma2
##       -0.8451  0.2868  0.2429
## s.e.   0.0286  0.0468  0.0510
##
## sigma^2 = 530.1:  log likelihood = -1931.66
## AIC=3871.32   AICc=3871.41   BIC=3887.51
##
## Training set error measures:
##                       ME     RMSE      MAE MPE MAPE      MASE        ACF1
## Training set -0.4074642 22.91533 16.27369 NaN  Inf 0.8049532 -0.04794138
```

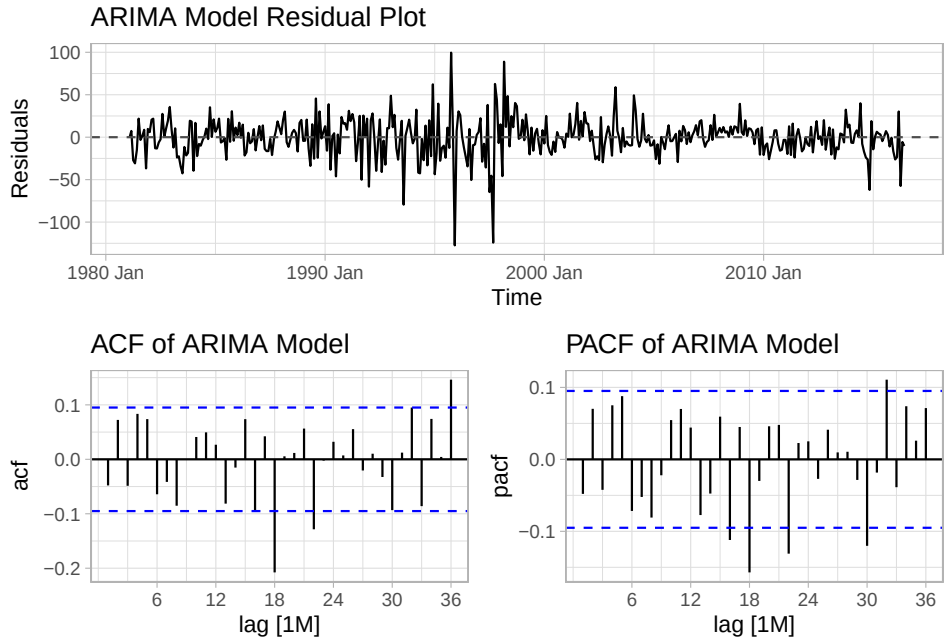
To illustrate the fit of the model, the following graph shows the original time series (black curve) and the model (blue curve).



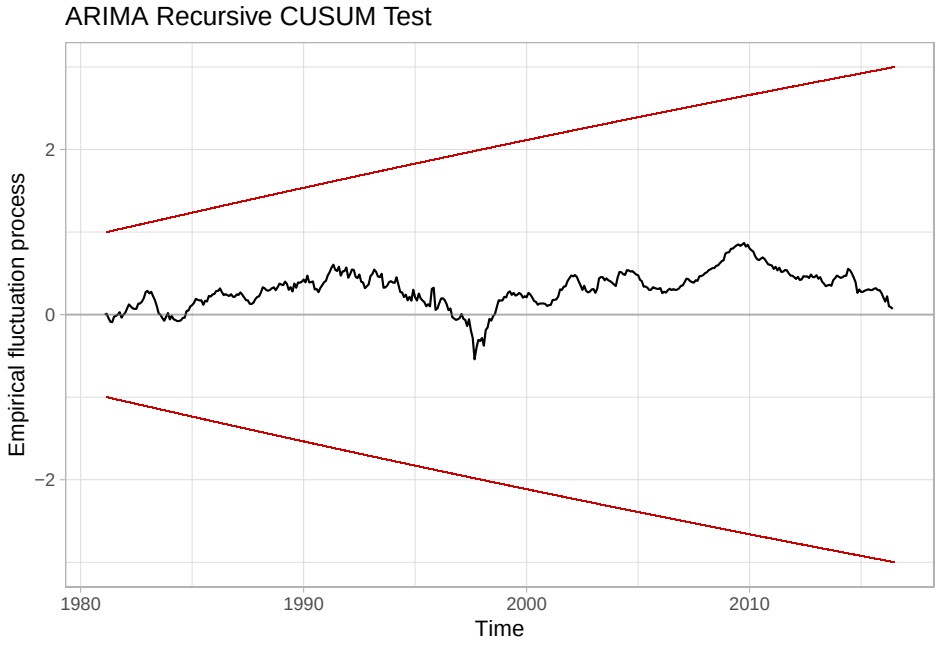
Assessing Model Validity

The time plot of the residuals still shows some structure, indicating that our ARIMA model might not have been fully effective at capturing the dynamics. The ACF shows multiple troubling lags, particularly lag 18,

which has a significant magnitude and indicates existing serial correlation. The PACF tells a similar story, with more lags having significant spikes. We will further analyze serial correlation when covering model diagnostic statistics.



Lastly, we look to the cumulative sum plot to identify any structural breaks. The CUSUM plot shows no significant divergence from 0, implying that there are no structural breaks in our ARIMA model.



Model Diagnostic Statistics

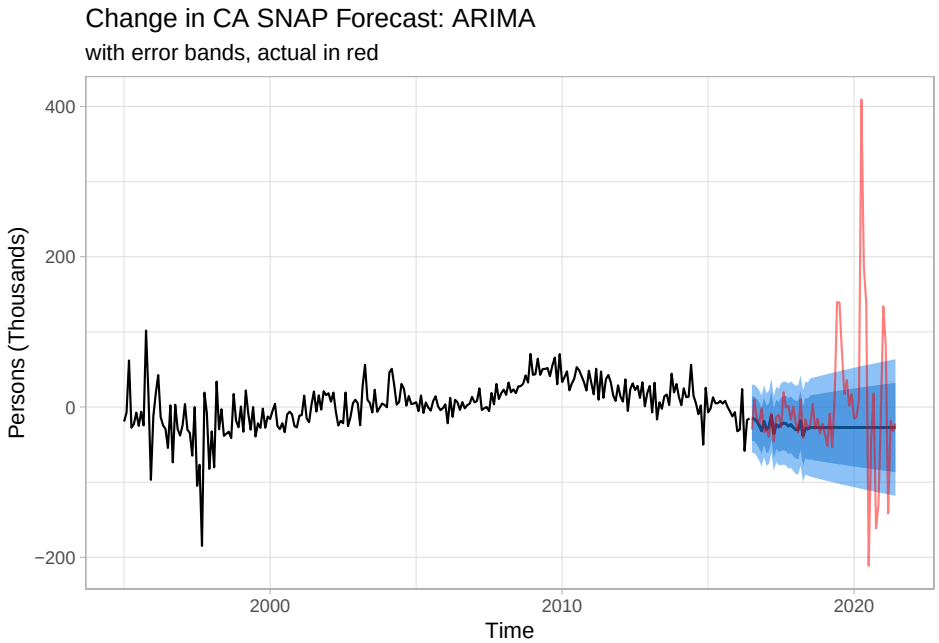
```
## --- Training Dataset ---
## MAE:  16.27369
## RMSE: 22.91533
## ME:   -0.4074642
```

```
##
##  Box-Ljung test
##
## data:  resids
## X-squared = 56.28, df = 24, p-value = 0.0002099
```

The ARIMA model has a very high MAE and RMSE, at 16.27369 and 22.91533, respectively, especially when compared to the sample standard deviation of about 30. Similar to our first model, these values are much too high for a suitable forecast, thus we do not expect this to perform well on the testing data. From the mean error (ME), we also can see that our model slightly overestimates the true values.

A Ljung-Box test on 24 lags resulted in a p-value of 0.0002099, so at 0.05 significance there is evidence of serial correlation in the residuals. This test further shows that the model is ineffective, as it did not wipe out the serial correlation.

Forecasting



In the plot above, we have a 5-year forecast using our ARIMA model for the time series with the true data overlaid in red. The model performs poorly throughout the forecast, only getting worse with the erratic behavior of the testing data, starting around 2019. This shortcoming is evident in the goodness-of-fit measures calculated for the testing set: the MAE and RMSE are very high, 48.44303 and 88.31337, respectively. And the ME shows that we underestimated the true behavior by a lot: 25.44684 on average.

```
## --- Testing Dataset ---
## MAE:  48.44303
## RMSE: 88.31337
## ME:   25.44684
```

(d) ETS Model

Model Construction

We used the `ets()` function in R to estimate an undamped ETS model with a Box-Cox transformation. We selected an undamped model because the time series does not exhibit any plateauing. The algorithm determined the optimized model to be an ETS(A,N,A), with the following state space equation formulations:

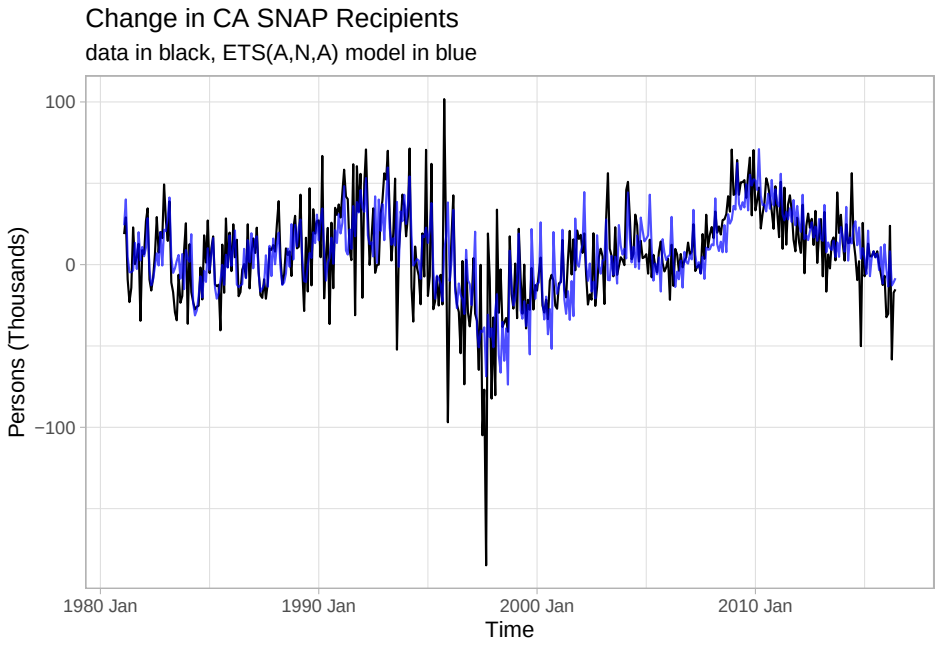
$$y_t = \ell_{t-1} + s_{t-12} + \varepsilon_t$$
$$\ell_t = \ell_{t-1} + \alpha\varepsilon_t$$
$$s_t = s_{t-12} + \gamma\varepsilon_t$$

$$\text{where} \quad \varepsilon_t \sim NID(0, \sigma^2)$$

The Box-Cox transformation parameter λ was found to be 0.9532 (close to 1, very little transformation occurred). The parameters (γ, α, σ) and initial values are summarized in the model summary below:

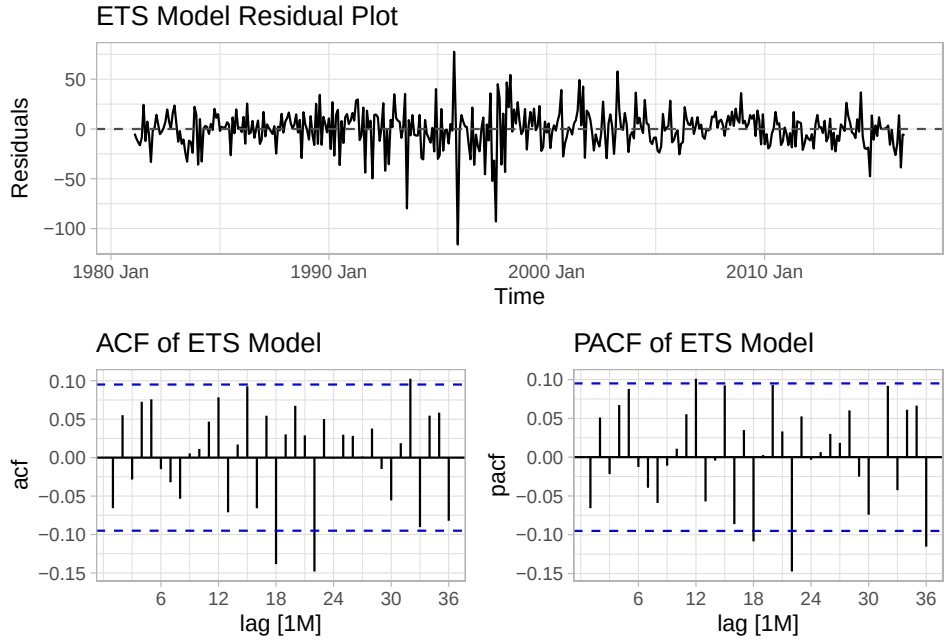
```
## ETS(A,N,A)
##
## Call:
##  ets(y = CA_ts, damped = FALSE, lambda = "auto")
##
##   Box-Cox transformation: lambda= 0.9532
##
##   Smoothing parameters:
##     alpha = 0.1554
##     gamma = 0.1539
##
##   Initial states:
##     l = 12.392
##     s = 3.0513 7.9693 -4.9283 9.8277 -9.9381 3.3356
##            -8.8243 -11.2318 -13.7285 -6.6276 22.6891 8.4054
##
##   sigma:  19.435
##
##      AIC     AICc      BIC
## 5109.918 5111.092 5170.700
##
## Training set error measures:
##                       ME     RMSE      MAE MPE MAPE      MASE        ACF1
## Training set -0.3814321 21.93345 15.34845 NaN  Inf 0.7591878 -0.06358461
```

To illustrate the fit of the model, the following graph shows the original time series (black curve) and the model (blue curve).

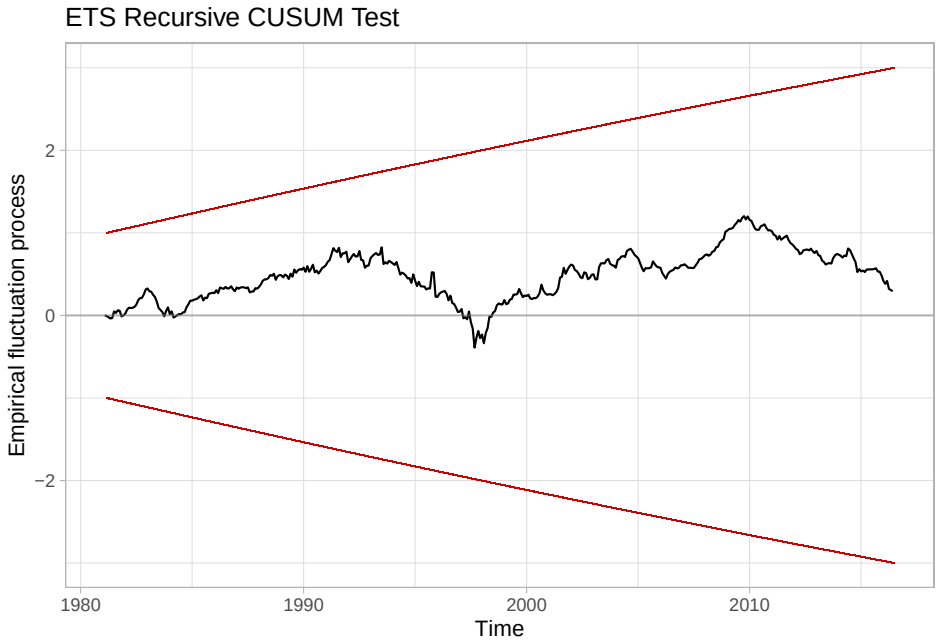


Assessing Model Validity

The time plot of the residuals still shows some structure, indicating that our ETS model might not have been fully effective at capturing the dynamics. The ACF shows two particularly troubling lags: lags 18 and 22, which both have significant magnitudes and indicate existing serial correlation. The PACF has lag 22 as a notably significant spike, also indicating a problem in the model. We will further analyze serial correlation when covering model diagnostic statistics.



Lastly, we look to the cumulative sum plot to identify any structural breaks. The CUSUM plot shows no significant divergence from 0, implying that there are no structural breaks in our ETS model.



Model Diagnostic Statistics

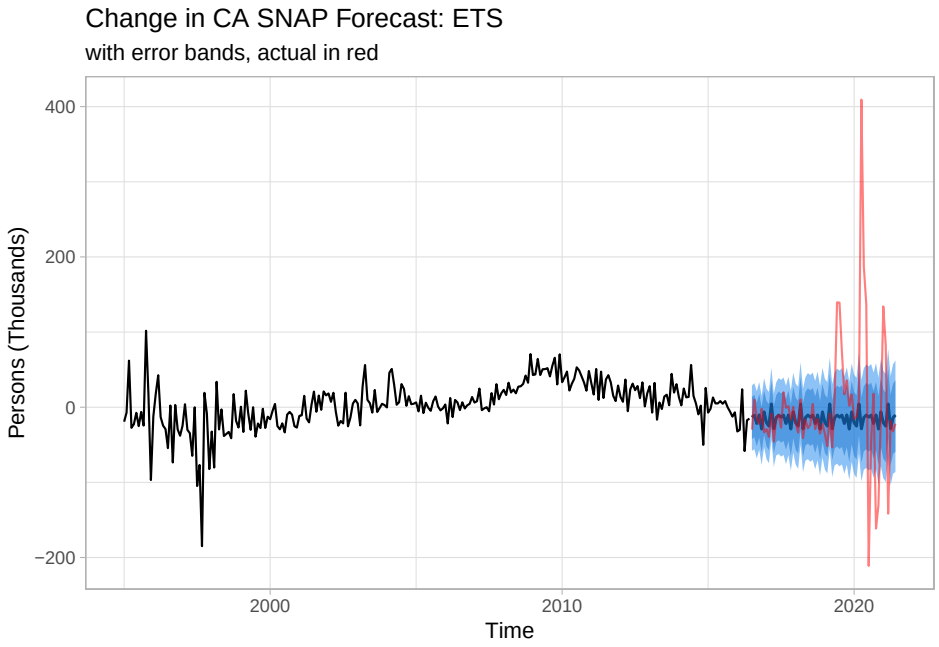
```
## --- Training Dataset ---
## MAE:  13.52962
## RMSE: 19.11226
## ME:   -0.3853634
```

```
##
##  Box-Ljung test
##
## data:  resids
## X-squared = 45.58, df = 24, p-value = 0.004971
```

The ETS model has a high MAE and RMSE, at 13.52962 and 19.11226, respectively, especially when compared to the sample standard deviation of approximately 30. Similar to our other models, these values are much too high for a suitable forecast, thus we do not expect this to perform well on the testing data. From the mean error (ME), we also can see that our model tends to overestimate the time series.

A Ljung-Box test on 24 lags resulted in a p-value of 0.004971, so at 0.05 significance there is evidence of serial correlation in the residuals. This test further shows that the model is ineffective, as it did not wipe out the serial correlation.

Forecasting



In the plot above, we have a 5-year forecast using an ETS model for the time series with the true data overlaid in red. The model performs poorly throughout the forecast, only getting worse with the erratic behavior of the testing data, starting around 2019. This shortcoming is evident in the goodness-of-fit measures calculated for the testing set: the MAE and RMSE are very high: 46.45434 and 86.91557, respectively. And the ME shows that we tended to underestimate the true behavior: 14.63586 on average.

```
## --- Testing Dataset ---
## MAE:  46.45434
## RMSE: 86.91557
## ME:   14.63586
```

(e) Holt-Winters Model

Model Construction

We used the `hw()` function in R to estimate an undamped Holt-Winters seasonal model. An additive seasonality component was used. The Holt-Winters model is outlined with the following state space equation formulations:

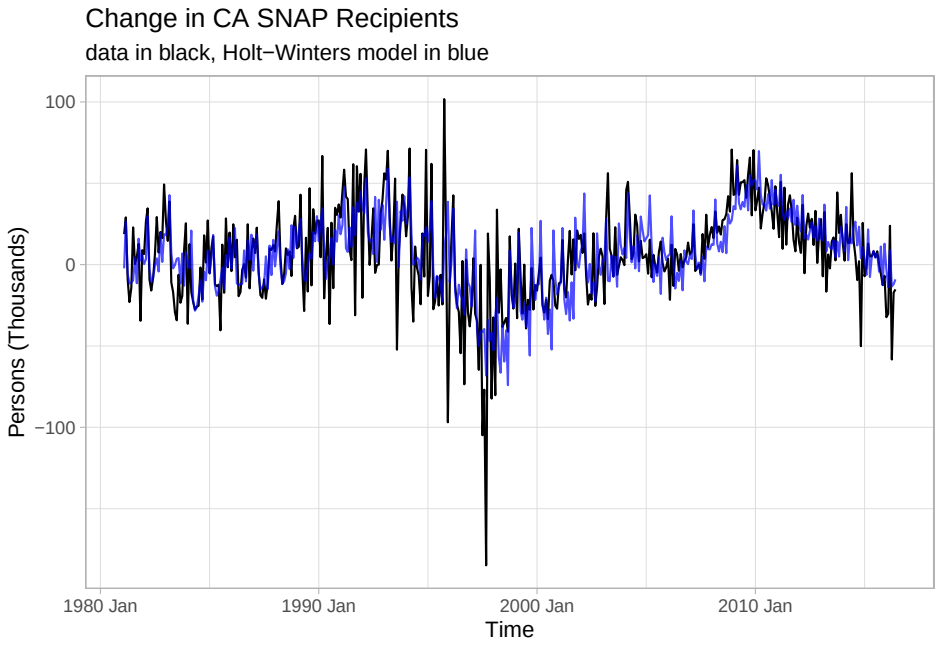
$$\begin{aligned}
\hat{y}_{t+h|t} &= \ell_t + hb_t + s_{t+h-12(k+1)} \\
\ell_t &= \alpha(y_t - s_{t-12}) + (1-\alpha)(\ell_{t-1} + b_{t-1}) \\
b_t &= \beta^*(\ell_t - \ell_{t-1}) + (1-\beta^*)b_{t-1} \\
s_t &= \gamma(y_t - \ell_{t-1} - b_{t-1}) + (1-\gamma)s_{t-m}
\end{aligned}$$

$$\text{where} \quad k = \lfloor (h-1)/m \rfloor$$

The parameters (α, β, γ) and initial values are summarized in the model summary below:

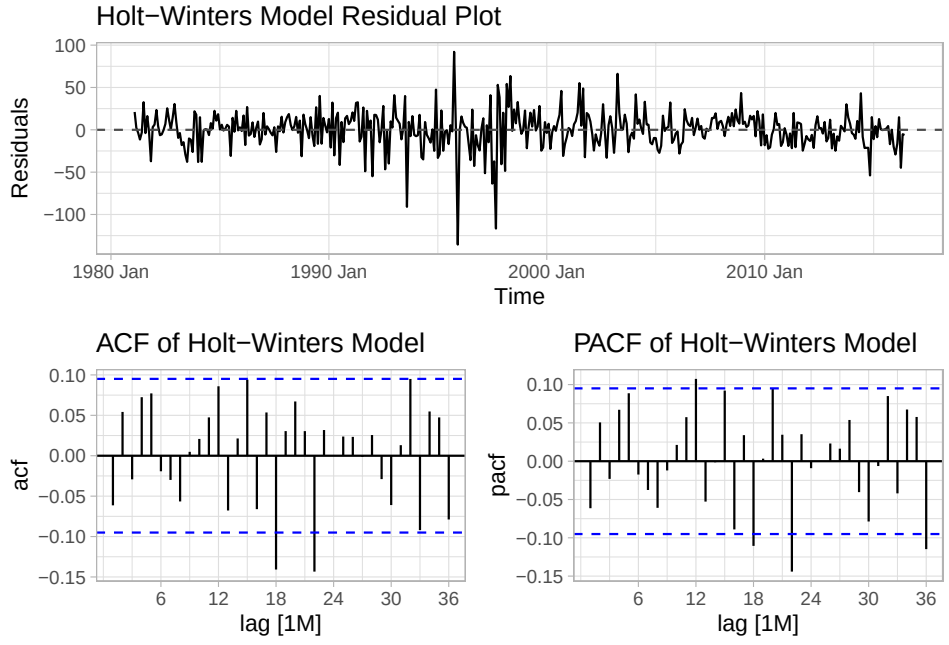
```
## Holt-Winters' additive method
##
## Call:
##  hw(y = CA_ts, h = 60, seasonal = "additive", damped = FALSE)
##
##   Smoothing parameters:
##     alpha = 0.1536
##     beta  = 1e-04
##     gamma = 0.1505
##
##   Initial states:
##     l = -2.9012
##     b = -0.0397
##     s = 1.6838 7.5139 -1.3376 11.4742 -13.4723 1.5509
##            -7.398 -9.3021 -11.609 -6.0549 26.3198 0.6314
##
##   sigma:  22.4949
##
##      AIC     AICc      BIC
## 5236.125 5237.628 5305.010
```

To illustrate the fit of the model, the following graph shows the original time series (black curve) and the model (blue curve).



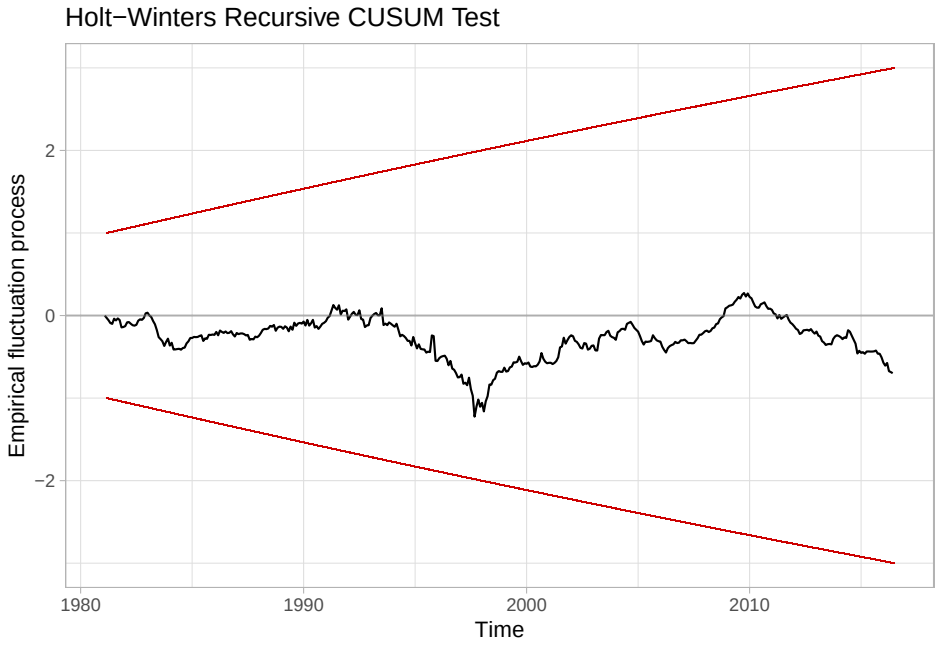
Assessing Model Validity

The residual plots of the Holt-Winters seasonal model show a similar behavior to that of the ETS model: some structure to the residuals, significant lags in the ACF at 18 and 22, and a significant lag at 22 in the PACF. All of these still indicate potential issues with the model. We will further analyze serial correlation when covering model diagnostic statistics.



Lastly, we look to the cumulative sum plot to identify any structural breaks. The CUSUM plot shows no

significant divergence from 0, implying that there are no structural breaks in our Holt-Winters model.



Model Diagnostic Statistics

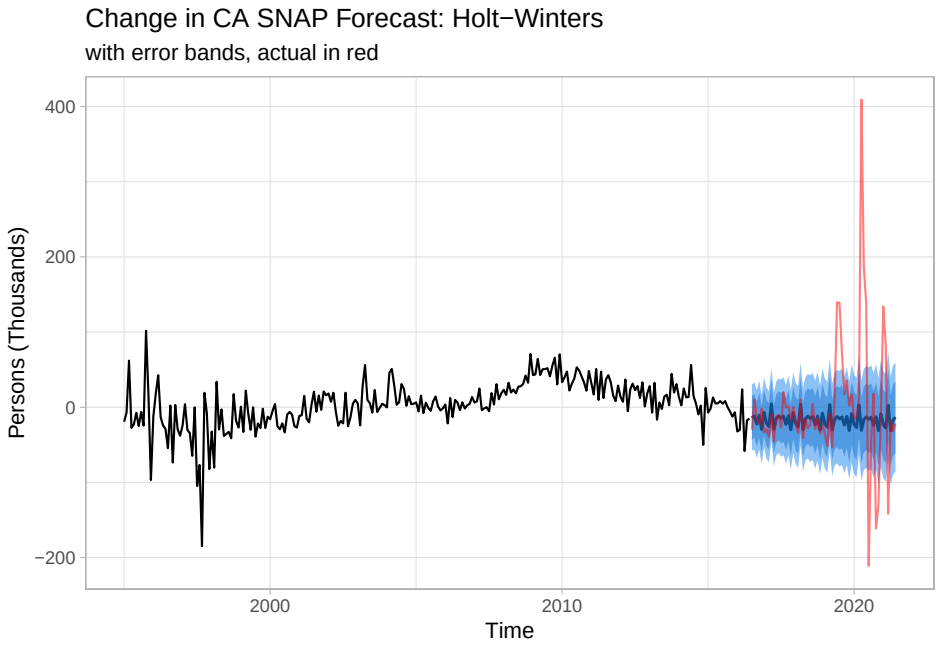
```
## --- Training Dataset ---
## MAE:  15.43868
## RMSE: 22.06741
## ME:   -0.003542301
```

```
##
##  Box-Ljung test
##
## data:  resids
## X-squared = 45.207, df = 24, p-value = 0.005505
```

The Holt-Winters model has a high MAE and RMSE, at 15.43868 and 22.06741, respectively, especially when compared to the sample standard deviation of approximately 30. Similar to our other models, these values are much too high for a suitable forecast, thus we do not expect this to perform well on the testing data. From the mean error (ME), we also can see that our model tends to slightly overestimate the time series.

A Ljung-Box test on 24 lags resulted in a p-value of 0.005505, so at 0.05 significance there is evidence of serial correlation in the residuals. This test further shows that the model is ineffective, as it did not wipe out the serial correlation.

Forecasting



In the plot above, we have a 5-year forecast using a Holt-Winters seasonal model for the time series with the true data overlaid in red. The model performs poorly throughout the forecast, only getting worse with the erratic behavior of the testing data, starting around 2019. This shortcoming is evident in the goodness-of-fit measures calculated for the testing set: the MAE and RMSE are very high: 46.55207 and 87.32159, respectively. And the ME shows that we tended to underestimate the true behavior: 16.26301 on average.

```
## --- Testing Dataset ---
## MAE:  46.55207
## RMSE: 87.32159
## ME:   16.26301
```

(f) TBATS Model

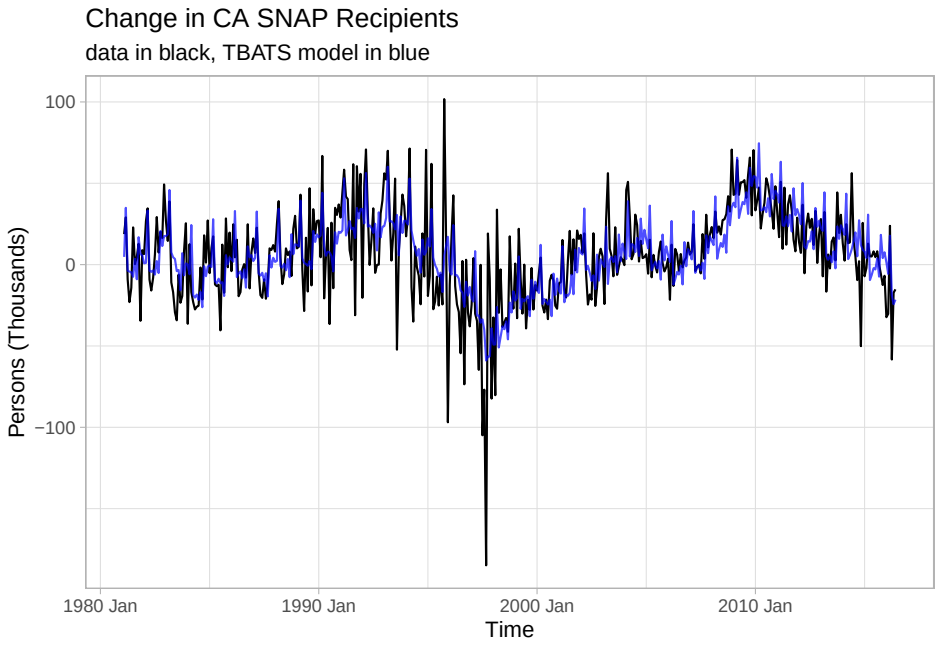
Model Construction

For our fifth model, we are using a model with trigonometric seasonality, a Box-Cox transformation, ARMA errors, trend and seasonality components, or a TBATS model for short. Since the model has a complex formulation of state space equations, we will not state them explicitly. The final model is of the form TBATS(1, {0,0}, -, {<12, 5>}).

The parameters and initial values are summarized in the model summary below:

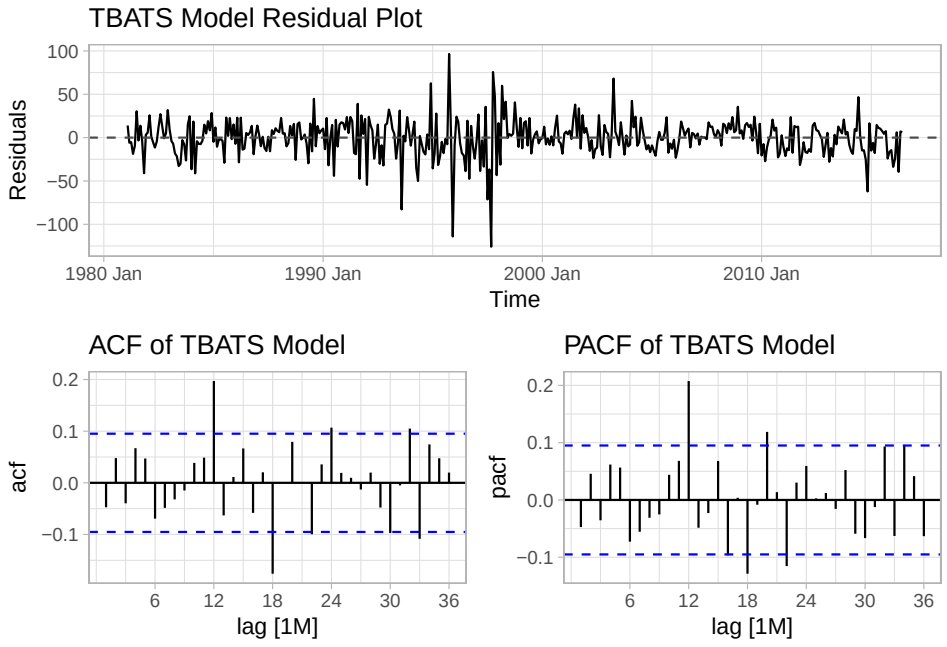
```
## TBATS(1, {0,0}, -, {<12,5>})
##
## Call: tbats(y = CA_ts)
##
## Parameters
##   Alpha: 0.1698578
##   Gamma-1 Values: -0.001137338
##   Gamma-2 Values: 0.002872805
##
## Seed States:
##            [,1]
##  [1,]  4.228998
##  [2,]  7.016725
##  [3,]  2.048242
##  [4,]  1.475069
##  [5,] -2.358124
##  [6,] -7.651214
##  [7,] -3.382341
##  [8,]  3.050391
##  [9,]  7.097613
## [10,]  2.838208
## [11,]  5.926808
##
## Sigma: 21.82811
## AIC: 5220.857
```

To illustrate the fit of the model, the following graph shows the original time series (black curve) and the model (blue curve).

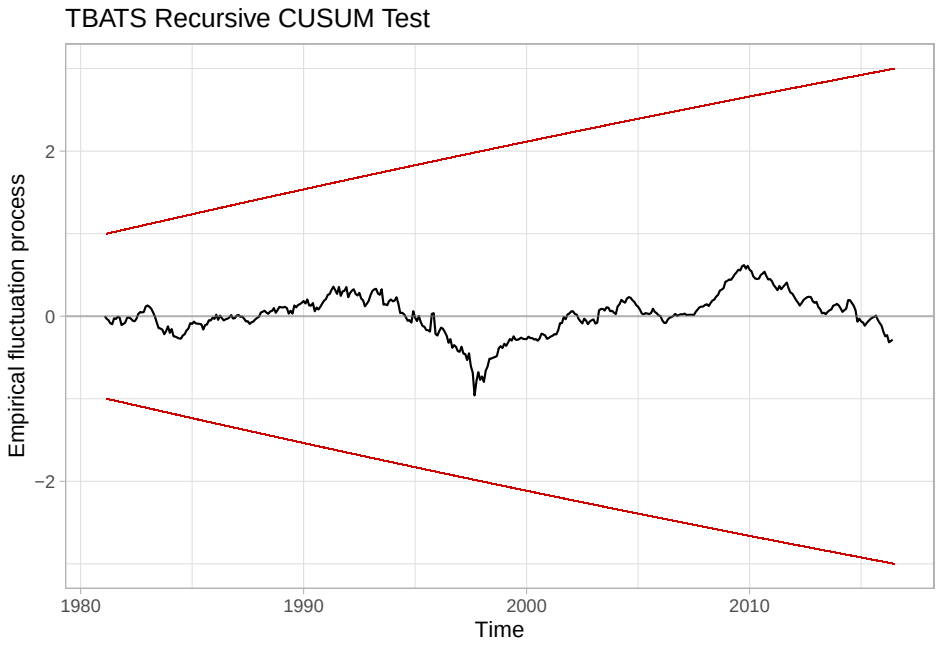


Assessing Model Validity

The residual plot of the TBATS model shows that not all structure was captured by the model. Unlike the previous models, this model's ACF and PACF show significant spikes at 12 and 18. These spikes still indicate issues in the model. We will further analyze serial correlation when covering model diagnostic statistics.



Lastly, we look to the cumulative sum plot to identify any structural breaks. The CUSUM plot shows no significant divergence from 0, implying that there are no structural breaks in our TBATS model.



Model Diagnostic Statistics

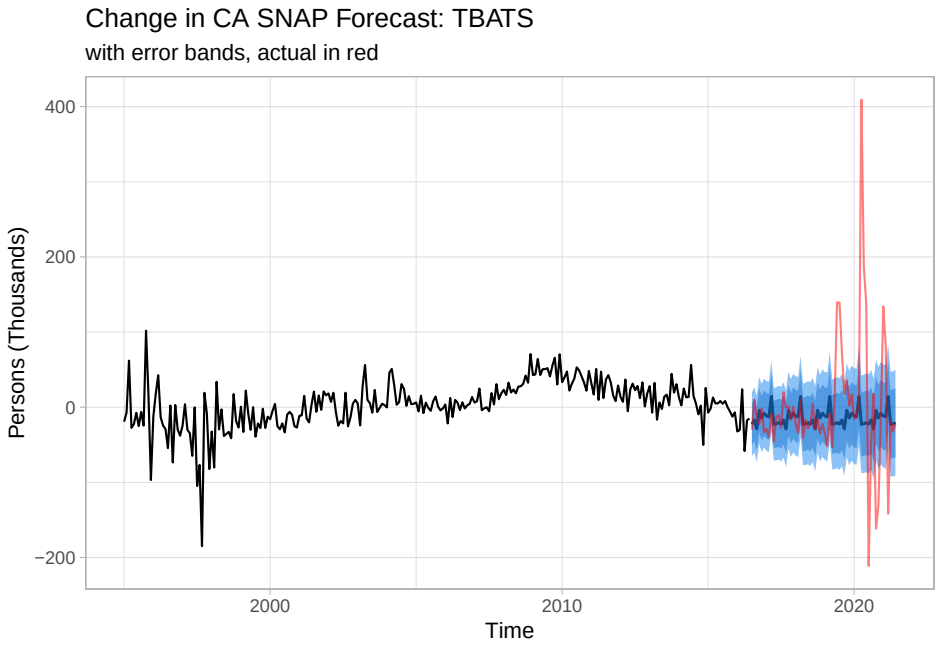
```
## --- Training Dataset ---
## MAE:  15.7007
## RMSE: 21.82811
## ME:   -0.2553708
```

```
##
##  Box-Ljung test
##
## data:  resids
## X-squared = 60.198, df = 24, p-value = 5.989e-05
```

The TBATS model has a high MAE and RMSE, at 15.7007 and 21.82811, respectively, especially when compared to the sample standard deviation of approximately 30. These values are much too high for a suitable forecast, thus we do not expect this to perform well on the testing data. From the mean error (ME), we also can see that our model tends to overestimate the time series.

A Ljung-Box test on 24 lags resulted in a p-value of 5.989×10^{-5}, so at 0.05 significance there is evidence of serial correlation in the residuals. This test further shows that the model is ineffective, as it did not wipe out the serial correlation.

Forecasting



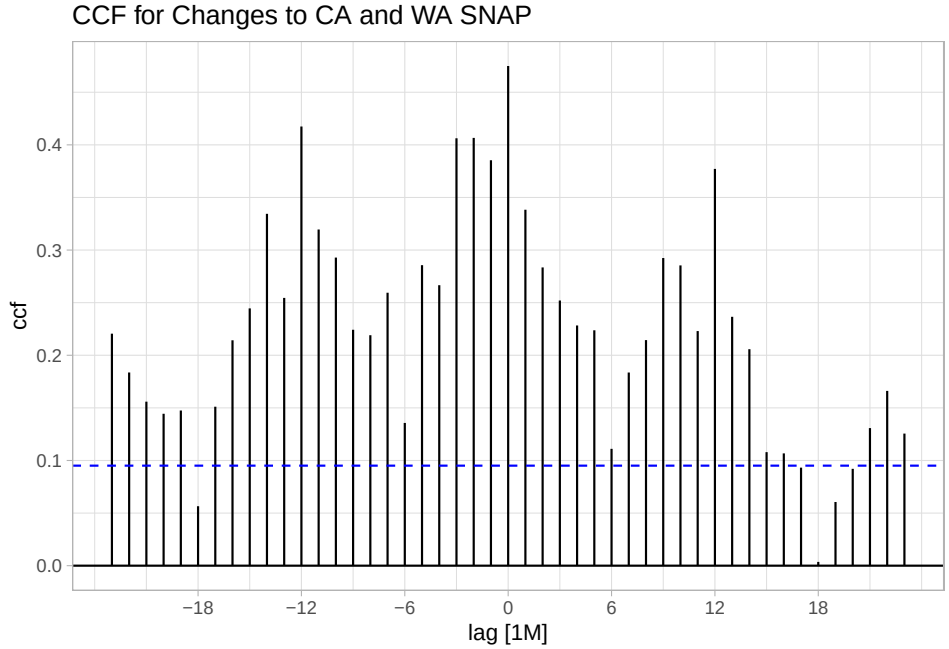
In the plot above, we have a 5-year forecast using our TBATS model for the time series with the true data overlaid in red. The model performs poorly throughout the forecast, only getting worse with the erratic behavior of the testing data, starting around 2019. This shortcoming is evident in the goodness-of-fit measures calculated for the testing set: the MAE and RMSE are very high: 48.07728 and 88.03457, respectively. And the ME shows that we tended to underestimate the true behavior: 13.37671 on average.

```
## --- Testing Dataset ---
## MAE:  48.07728
## RMSE: 88.03457
## ME:   13.37671
```

(g) VAR Model

VAR Order and Causality

For our sixth model, we attempt to model changes in California SNAP recipients using changes in Washington SNAP recipients in a vector autoregression model (VAR). To get an idea of the dynamics between the two series, we first look at the cross-correlation function plot below. While the strongest lags are on multiples of 12, there do seem to be particularly strong lags for -1, -2, and -3. This is indicative of a possible causal relationship between the two series that we will explore further.



To determine the order of the VAR model, we used the `VARselect()` function. We determined an order of 2 since that was chosen as having the lowest BIC.

Table 3: VAR lag-order criteria

AIC	HQ	BIC	FPE
10	9	2	10

Next, we want to establish Granger-causality at the second order between the two time series, preferably changes to Washington SNAP recipients "causing" changes to California.

```
## ----- WA cause CA -----

## Granger causality test
##
## Model 1: CA_ts ~ Lags(CA_ts, 1:2) + Lags(WA_ts, 1:2)
## Model 2: CA_ts ~ Lags(CA_ts, 1:2)
##   Res.Df Df      F    Pr(>F)
## 1    418
## 2    420 -2 15.656 2.776e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

##
##
```

```
## ----- CA cause WA -----
## Granger causality test
##
## Model 1: WA_ts ~ Lags(WA_ts, 1:2) + Lags(CA_ts, 1:2)
## Model 2: WA_ts ~ Lags(WA_ts, 1:2)
##   Res.Df Df      F Pr(>F)
## 1    418
## 2    420 -2 1.7833 0.1694
```

From the above code output, we observe that California does not have a significant causal effect on Washington; however, Washington has a significant causal effect on California at the 0.001 level (success!). Next we will move on to fitting the VAR model.

Model Construction

In general, the VAR model of order 2 takes the following form:

$$y_{1,t} = \beta_{1,1} \times y_{1,t-1} + \beta_{1,2} \times y_{1,t-2} + \beta_{2,1} \times y_{2,t-1} + \beta_{2,2} \times y_{2,t-2} + \varepsilon_{1,t}$$
$$y_{2,t} = \beta_{2,1} \times y_{2,t-1} + \beta_{2,2} \times y_{2,t-2} + \beta_{1,1} \times y_{1,t-1} + \beta_{1,2} \times y_{1,t-2} + \varepsilon_{2,t}$$

For the model of California's SNAP changes (Washington is exogenous), all lags of both series are significant at the 0.01 significance level. As for the model of Washington's SNAP changes (California is exogenous), only the lags of Washington are significant at the 0.05 significance level; all the lags for California are not significant at 5%. The coefficients for both model are summarized in the tables below. Note that we will be using the first model, where California is the endogenous variable.

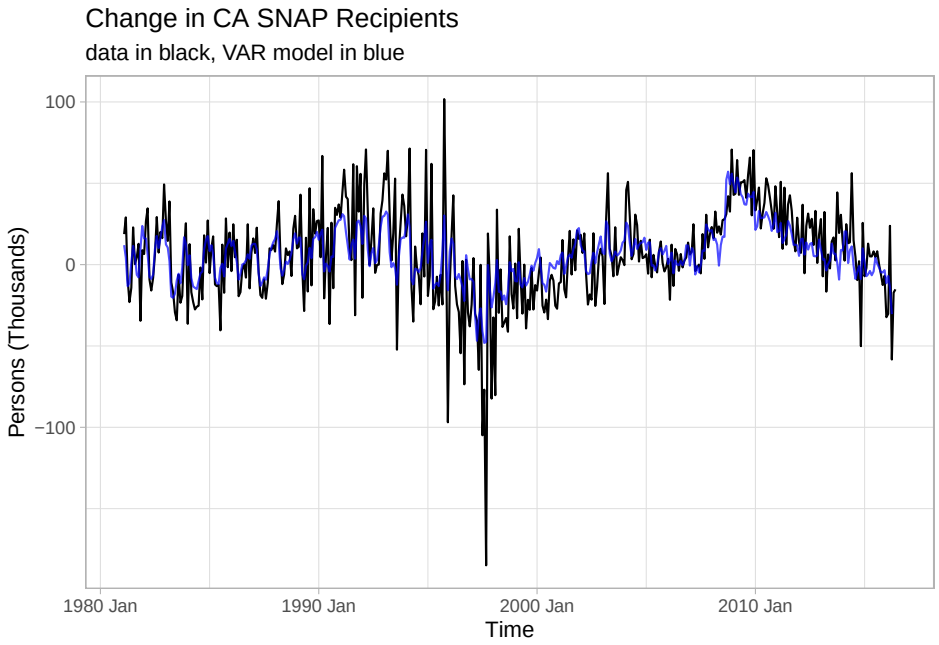
Table 4: California VAR Model Estimates

	CA L1	WA L1	CA L2	WA L2	Const
Coefficients	0.12953	0.61838	0.23641	0.63359	1.74717

Table 5: Washington VAR Model Estimates

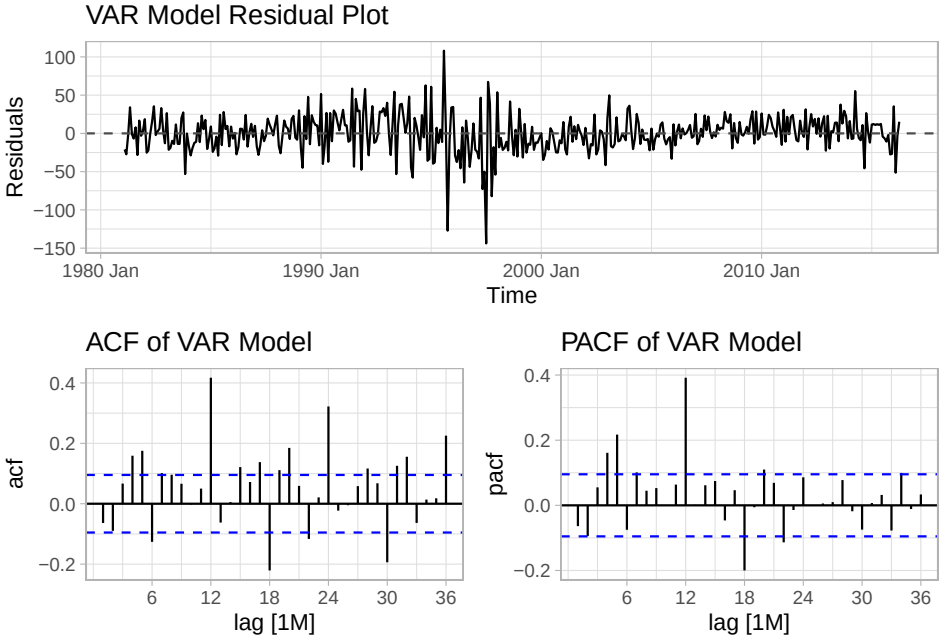
	CA L1	WA L1	CA L2	WA L2	Const
Coefficients	0.02312	0.27327	0.00433	0.3087	0.50795

To illustrate the fit of the model, the following graph shows the original time series (black curve) and the model (blue curve).

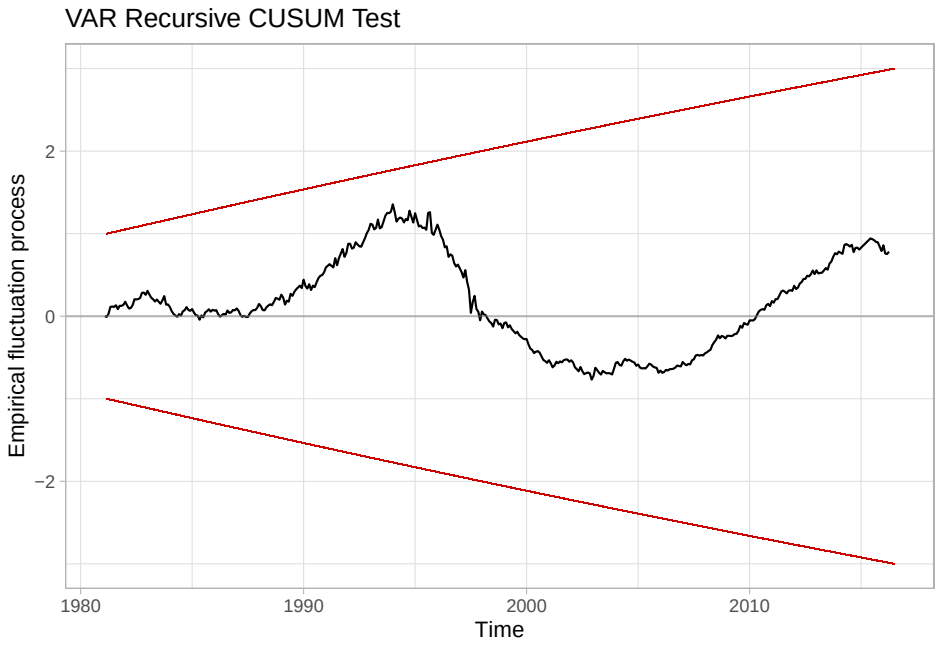


Assessing Model Validity

The residual plot of the VAR model shows that much of the structure in the residuals is still present. The ACF and PACF have lags 12 and 18 as significant, with the former reaching to almost 0.4, which is very high. There are other significant lags in the ACF too, so we ultimately conclude that there is still serial correlation in the residuals.



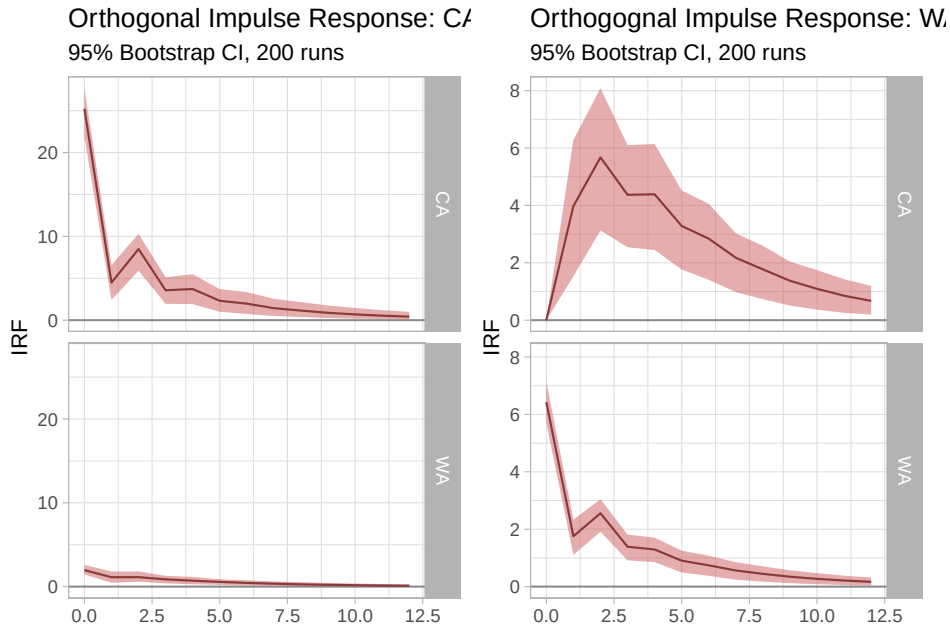
The cumulative sum of the residuals shows fluctuations around 0, though there is no significant divergence. Thus there are not any structural breaks in the model.



Lastly, we will look at the impulse response functions for the VAR model to support our assumption of "causality." The most important plots are the effect of Washington on California (top right) and the effect of California on Washington (bottom left) since the other two plots are just the self-effect of a shock.

The bottom left plot shows very little movement away from 0, supporting our earlier claim about California not "causing" Washington.

The top right plot shows a start from 0 and a significant rising over the next two months. The impulse response function peaks at just under 6 at the second month and begins to fall; even after a year the effect is falling but not 0. From this plot, we see that a shock to Washington leads to a significant and lasting effect on California, supporting the claim that Washington "causes" California, and we are therefore justified in using Washington to model California.



Model Diagnostic Statistics

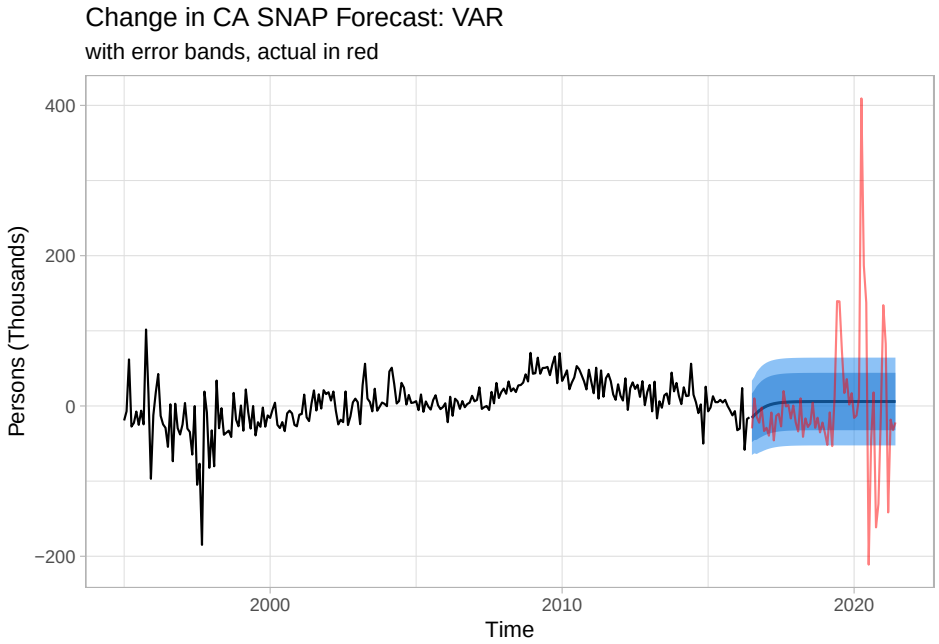
```
## --- Training Dataset ---
## MAE:  18.58397
## RMSE: 25.08801
## ME:   2.347479e-15
```

```
##
##  Box-Ljung test
##
## data:  resids
## X-squared = 241.52, df = 24, p-value < 2.2e-16
```

The VAR model has a high MAE and RMSE, at 18.58397 and 25.08801, respectively, especially when compared to the sample standard deviation of approximately 30. These values are much too high for a suitable forecast, thus we do not expect this to perform well on the testing data. Interestingly, the mean error of the VAR model is approximately 0, with just a slight amount of underestimation from our model.

A Ljung-Box test on 24 lags resulted in a p-value of essentially 0, so at 0.05 significance there is strong evidence of serial correlation in the residuals. This test further shows that the model is ineffective, as it did not wipe out the serial correlation.

Forecasting



In the plot above, we have a 5-year forecast using our VAR model for the time series with the true data overlaid in red. The model generally captures the true behavior within its error bands until around 2019, when the time series diverges significantly. We note that this VAR forecast – like standard AR forecasts – shows some movement from the final observation, but eventually converges to the mean.

The model's shortcomings are evident in the goodness-of-fit measures calculated for the testing set: the

MAE and RMSE are very high: 50.64799 and 84.43477, respectively. And the ME shows that we tended to overestimate the true behavior by 4.863295 on average.

```
## --- Testing Dataset ---
## MAE:  50.64799
## RMSE: 84.43477
## ME:   -4.863295
```

(h) Prophet Model

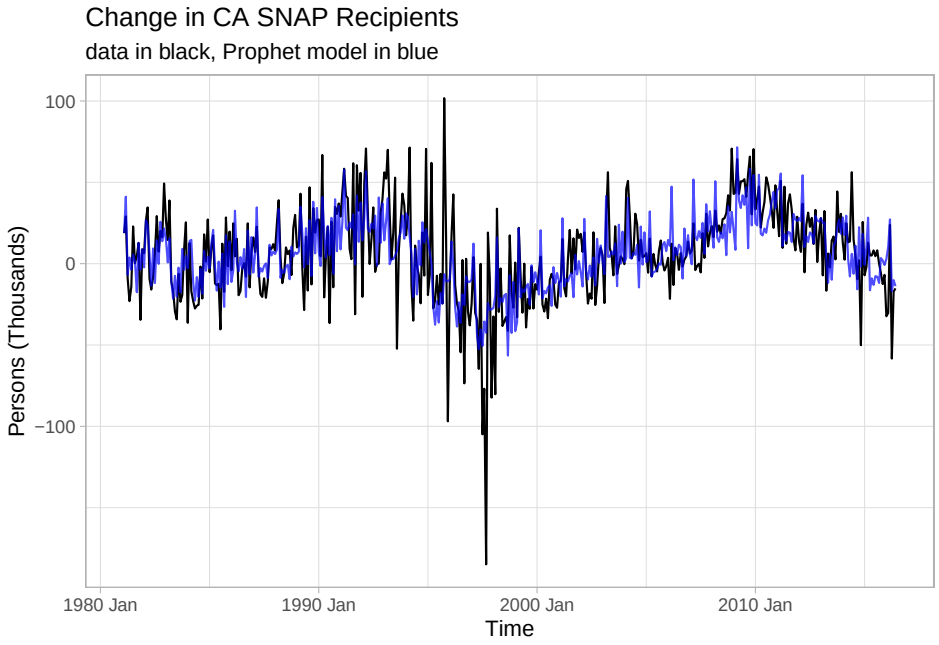
Model Construction

For our seventh model, we are using a Prophet model. Generally the model takes the following (nonlinear) form:

$$y_t = g(t) + s(t) + h(t) + \varepsilon_t$$

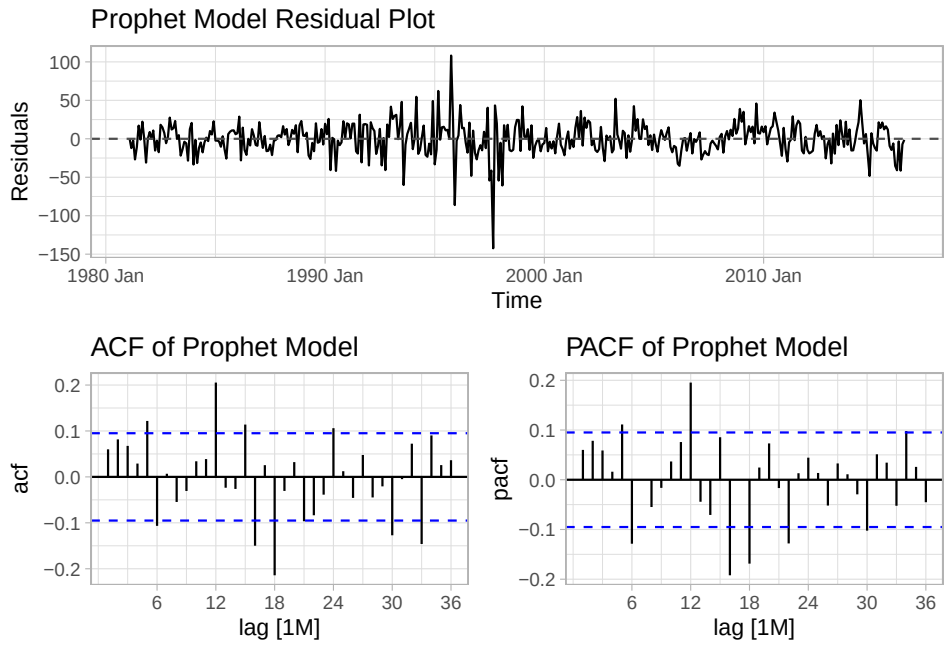
Here $g(t)$ is the growth term, $s(t)$ captures seasonal patterns, and $h(t)$ describes holiday effects. The parameters of the model were algorithmically determined.

To illustrate the fit of the model, the following graph shows the original time series (black curve) and the model (blue curve).

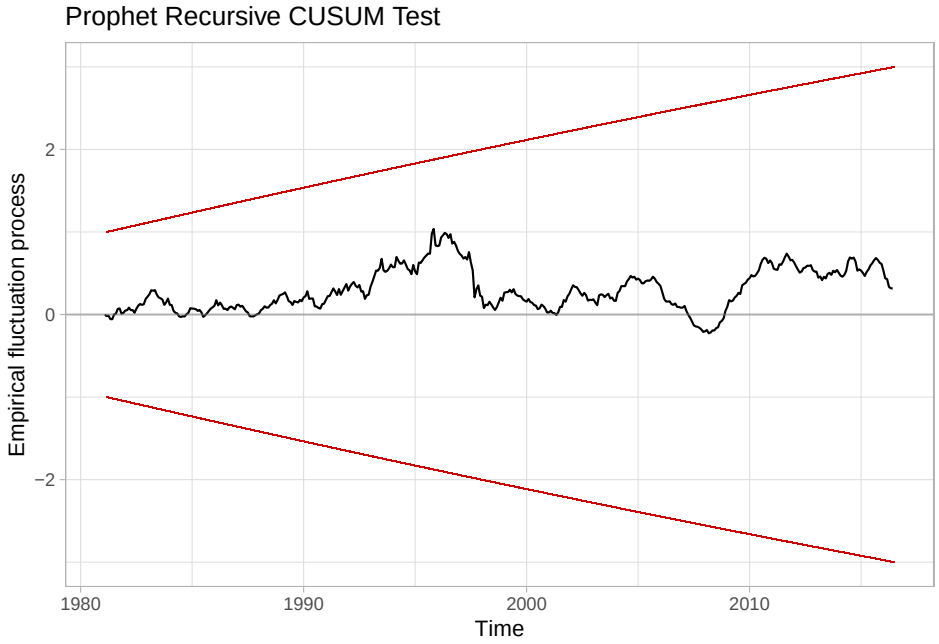


Assessing Model Validity

The residual plot of the Prophet model shows that there is still some remaining structure to the residuals, indicating shortcomings of out model. The ACF and PACF both have multiple significant lags, with the three highest being lag 12, 16, and 18. The significant lags indicate the presence of serial correlation in the residuals.



Lastly, we look to the cumulative sum plot to identify any structural breaks. The CUSUM plot shows no significant divergence from 0, implying that there are no structural breaks in our Prophet model.



Model Diagnostic Statistics

```
## --- Training Dataset ---
## MAE:  15.39964
## RMSE: 21.20363
```

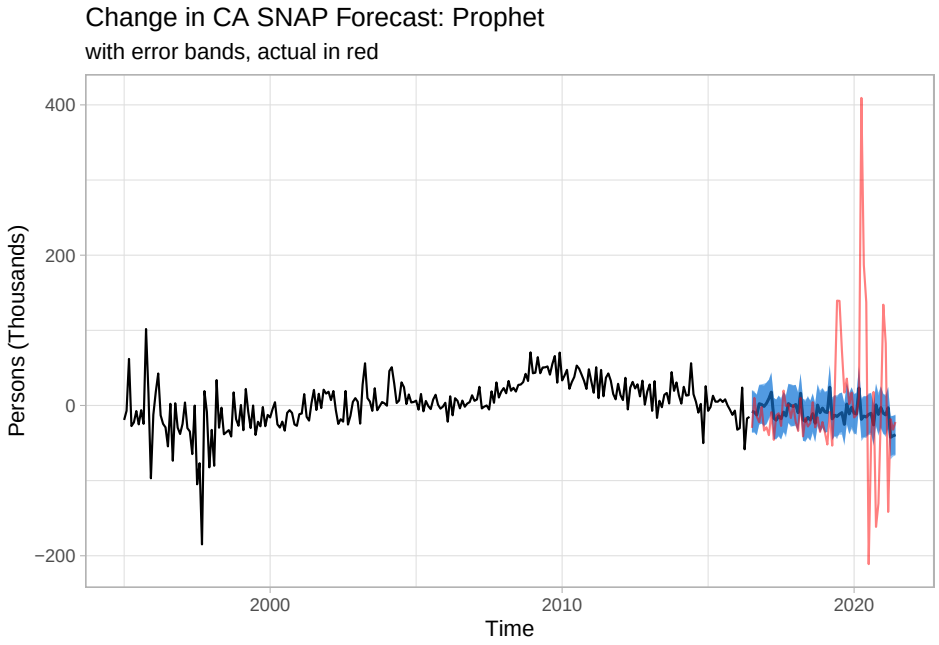
```
## ME:   0.07431906
```

```
##
##  Box-Ljung test
##
## data:  resids
## X-squared = 90.46, df = 24, p-value = 1.21e-09
```

The Prophet model has a high MAE and RMSE, at 15.39964 and 21.20363, respectively, especially when compared to the sample standard deviation of approximately 30. From the mean error (ME), we also can see that our model tends to underestimate the time series.

A Ljung-Box test on 24 lags resulted in a p-value of 1.21×10^{-9}, so at 0.05 significance there is evidence of serial correlation in the residuals. This test further shows that the model is ineffective, as it did not wipe out the serial correlation.

Forecasting



In the plot above, we have a 5-year forecast using our Prophet model for the time series with the true data overlaid in red. The model performs fairly well until around 2019 when the data diverges. This shortcoming is evident in the goodness-of-fit measures calculated for the testing set: the MAE and RMSE are very high: 49.20302 and 87.06276, respectively. And the ME shows that we tended to underestimate the true behavior: 8.411579 on average.

```
## --- Testing Dataset ---
## MAE:  49.20302
## RMSE: 87.06276
## ME:   8.411579
```

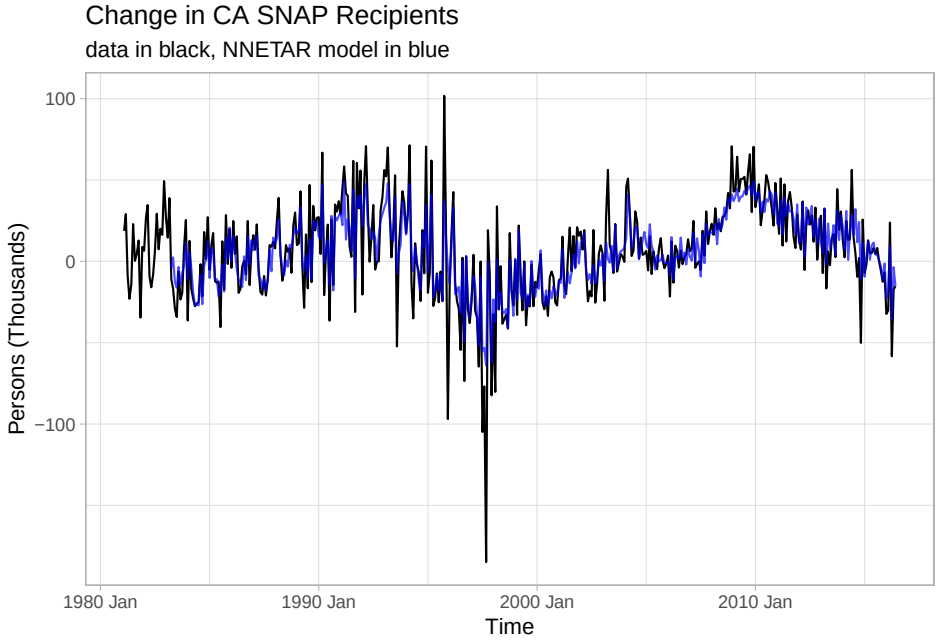
(i) NNETAR Model

Model Construction

For our eighth model, we are using a neural network autoregression, or NNETAR. The machine learning algorithm for building this model uses deep learning to construct autoregressive connections; for that reason we will not be able to extract or characterize the model beyond describing it as a variant of an AR model. We algorithmically determined $k = 3$ to be the optimal parameter; thus our model takes the form NNAR(26,1,3)[12]: 3 for the number of neurons in the hidden layer and 12 for monthly observations. The model is analogous to an ARIMA(26,0,0)(1,0,0)[12] model but with nonlinear functions. The NNETAR model is summarized in the code output below.

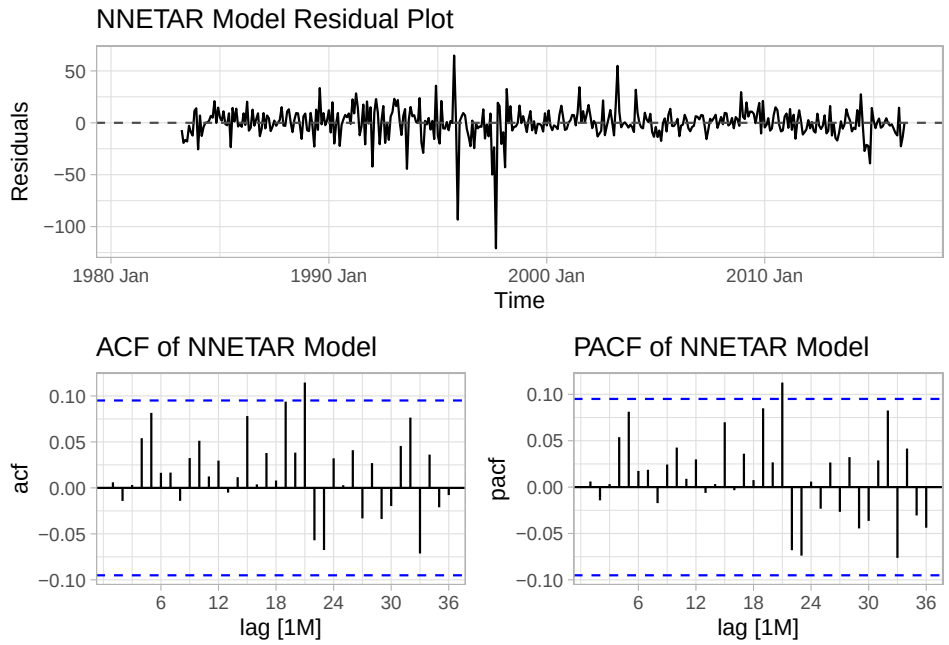
```
## Series: CA_ts
## Model:  NNAR(26,1,3)[12]
## Call:   nnetar(y = CA_ts, size = 3)
##
## Average of 20 networks, each of which is
## a 26-3-1 network with 85 weights
## options were - linear output units
##
## sigma^2 estimated as 219.7
```

To illustrate the fit of the model, the following graph shows the original time series (black curve) and the model (blue curve).

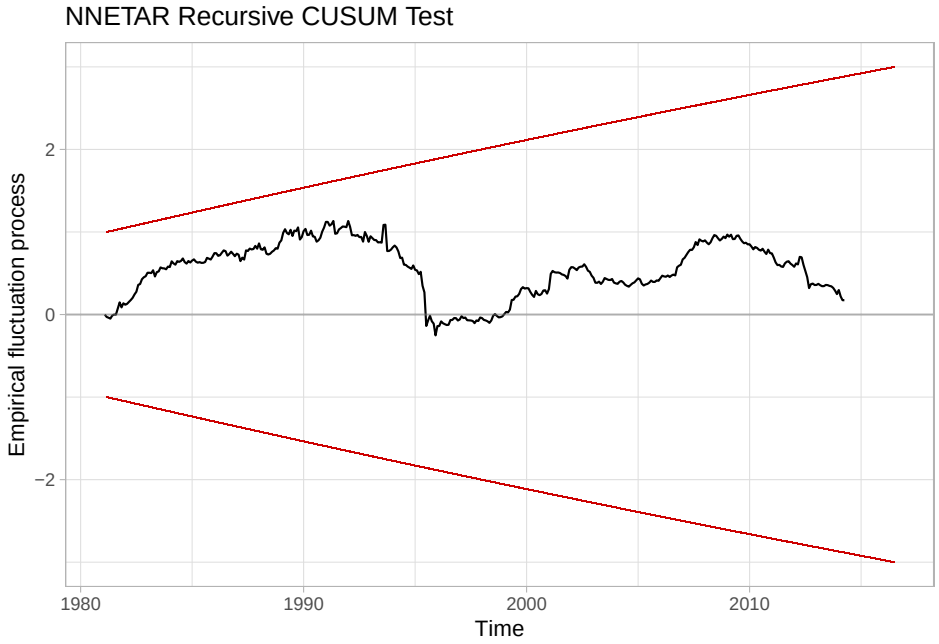


Assessing Model Validity

The residual plot for the NNETAR model shows very little structure, possibly displaying a white noise process. The ACF and PACF both have only a few lags being significant, but they have a small enough magnitude such that they are not significant after all. We would venture to say that this model *was* successful in eliminating serial correlation.



Lastly, we look to the cumulative sum plot to identify any structural breaks. The CUSUM plot shows no significant divergence from 0, implying that there are no structural breaks in our TBATS model.



Model Diagnostic Statistics

```
## --- Training Dataset ---
## MAE:  9.654533
## RMSE: 14.82094
```

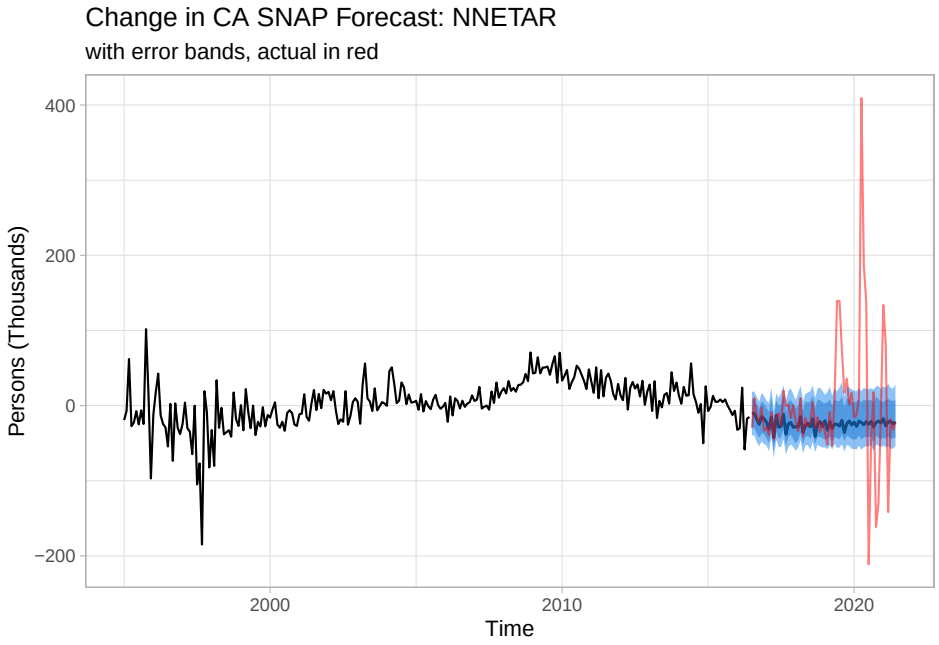
```
## ME:   -0.004704824
```

```
##
##  Box-Ljung test
##
## data:  resids
## X-squared = 23.12, df = 24, p-value = 0.5127
```

The NNETAR model has a comparatively good MAE and RMSE, at 8.711067 and 13.59463, respectively. From the mean error (ME), we also can see that our model tends to underestimate the time series.

A Ljung-Box test on 24 lags resulted in a p-value of 0.5916, so at 0.05 significance there not enough evidence to indicate serial correlation. Therefore the NNETAR model was effective at capturing the structures from the time series.

Forecasting



In the plot above, we have a 5-year forecast using our NNETAR model for the time series with the true data overlaid in red. After around 2019, the model performs fully, as seen by the goodness-of-fit measures calculated for the testing set: the MAE and RMSE are very high, 46.41403 and 85.07371, respectively. And the ME shows that we tended to underestimate the true behavior: 9.055539 on average.

```
## --- Testing Dataset ---
## MAE:  47.42223
## RMSE: 87.42868
## ME:   23.17299
```

(j) Combined Model

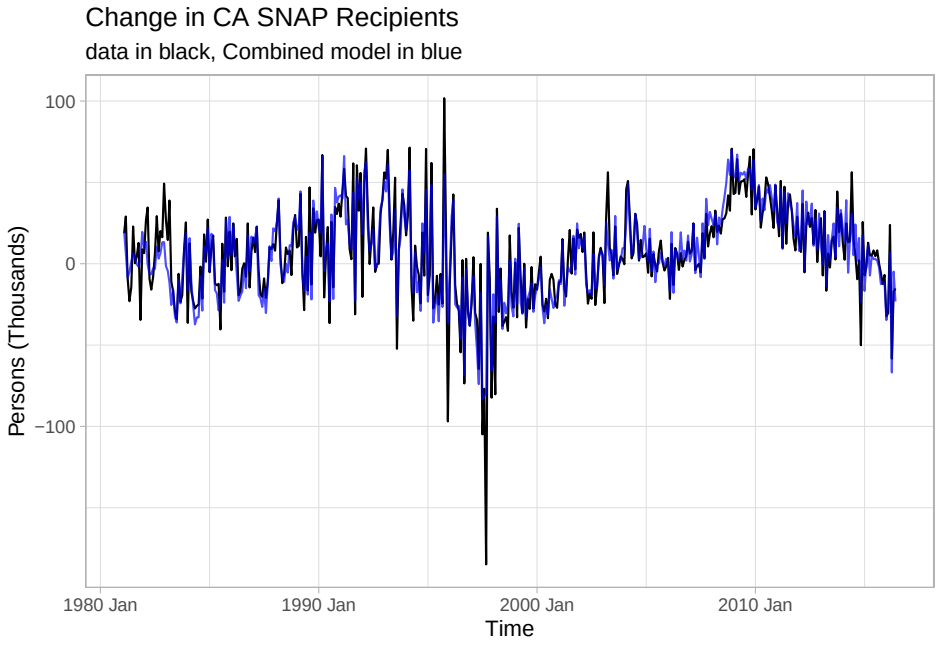
Model Construction

To create a combined model of the previous 8 models, we opted to use a static weighting scheme with weights determined with a linear regression. The weights are summarized in the table below.

Table 6: Weights for Combined Model

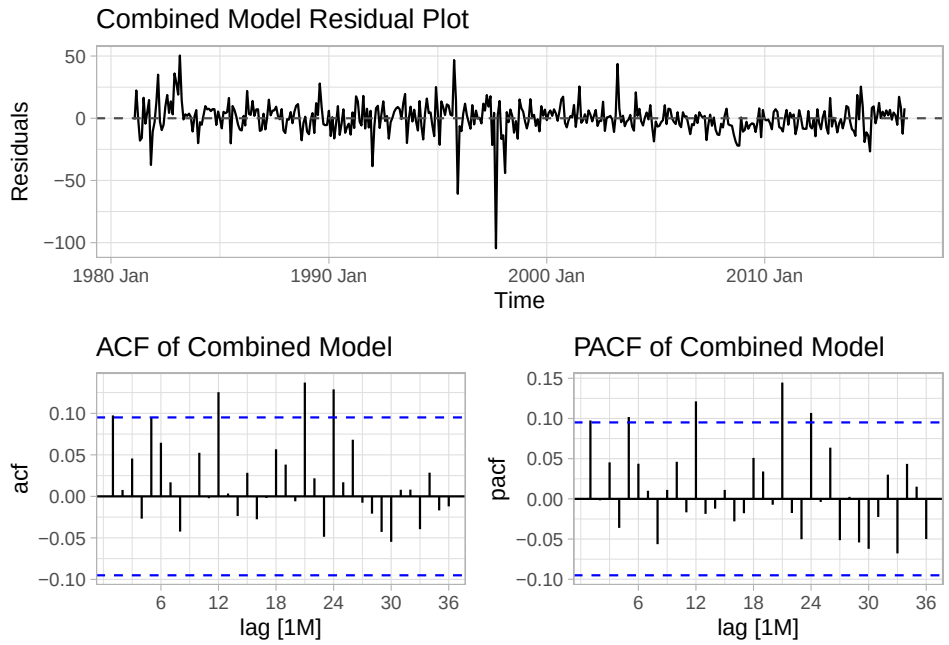
STL+ARMA	ARIMA	ETS	Holt-Winters	TBATS	VAR	Prophet	NNETAR
0.20538	-0.35061	0.62901	-0.90409	0.08172	0.39135	0.0993	1.21279

To illustrate the fit of the model, the following graph shows the original time series (black curve) and the model (blue curve).

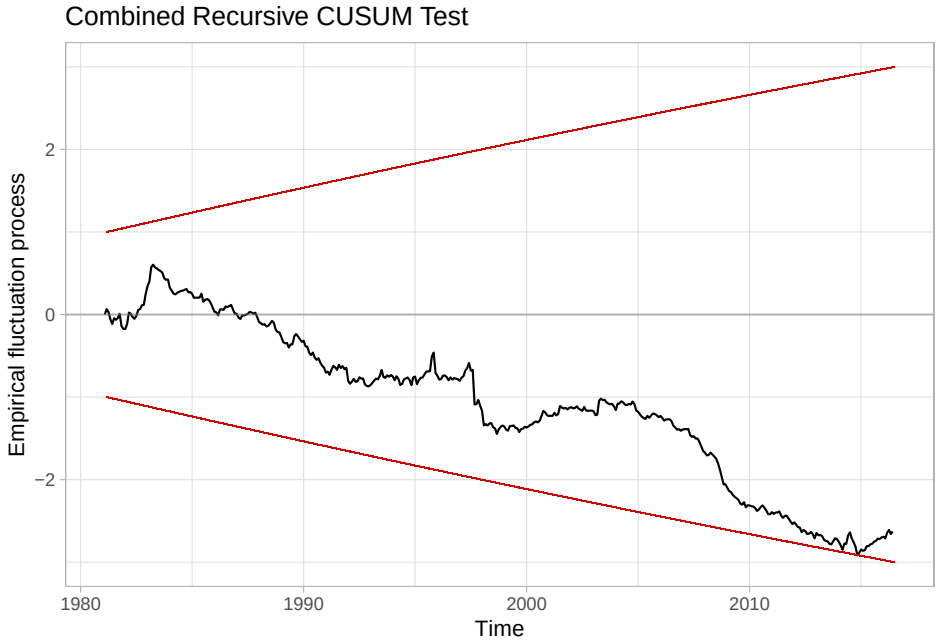


Assessing Model Validity

The residual plot of the combined model shows that much of the structure was captured by the model. This model's ACF and PACF have some significant spikes (like at lag 21 on both plots). Since the significant spikes have a low magnitude, a visual analysis is inconclusive so a formal statistical test will be necessary.



Lastly, we look to the cumulative sum plot to identify any structural breaks. The CUSUM plot shows no significant divergence from 0, implying that there are no structural breaks in our combined model.



Model Diagnostic Statistics

```
## --- Training Dataset ---
## MAE:  8.386259
## RMSE: 12.32178
```

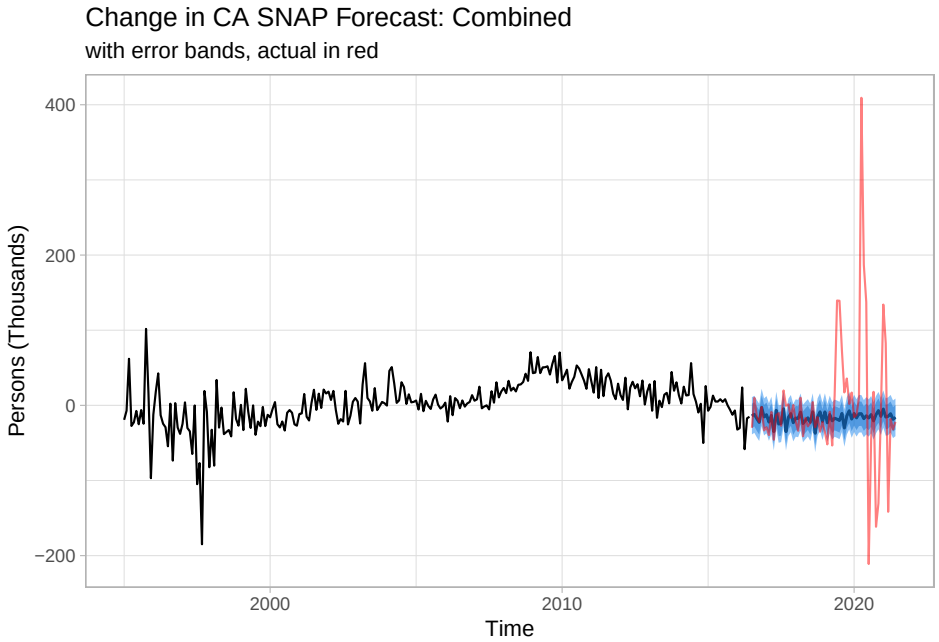
```
## ME:   1.446163e-15
```

```
##
##  Box-Ljung test
##
## data:  resids
## X-squared = 40.274, df = 24, p-value = 0.01998
```

The TBATS model has very good MAE and RMSE, at 8.151137 and 11.97867, respectively, compared to its component models. These values are much too high for a suitable forecast, thus we do not expect this to perform well on the testing data. From the mean error (ME), we also can see that our model has almost a zero mean error (likely due to the least squares regression).

A Ljung-Box test on 24 lags resulted in a p-value of 0.01091, so at 0.05 significance there is evidence of serial correlation in the residuals. This test shows that the model is ineffective, as it did not wipe out the serial correlation.

Forecasting



In the plot above, we have a 5-year forecast using our combined model for the time series with the true data overlaid in red. The model performs relatively well at the beginning of the forecast, but performs worse with the erratic behavior of the testing data starting around 2019. This shortcoming is evident in the goodness-of-fit measures calculated for the testing set: the MAE and RMSE are very high: 45.11352 and 85.87429, respectively. And the ME shows that we tended to underestimate the true behavior: 16.61062 on average.

```
## --- Testing Dataset ---
## MAE:  46.07908
## RMSE: 85.52361
## ME:   15.46789
```

(k) Model Comparison

To compare the nine models build above, we will rank them by their RMSE on the test data set. The results can be seen in the table below.

Table 7: Goodness-of-fit Measures

	MAE	RMSE	ME
VAR	50.648	84.435	-4.863
Combined	46.079	85.524	15.468
ETS	46.454	86.916	14.636
Prophet	49.203	87.063	8.412
Holt-Winters	46.552	87.322	16.263
NNETAR	47.422	87.429	23.173
TBATS	48.077	88.035	13.377
ARIMA	48.443	88.313	25.447
STL+ARMA	46.295	88.742	18.380

Interestingly, the model with the lowest RMSE is the VAR model, whose forecast converged to the mean over time and did not account for more complex dynamics. Despite having the lowest RMSE, the VAR model had the highest MAE, meaning that we might not consider this the best performing model overall.

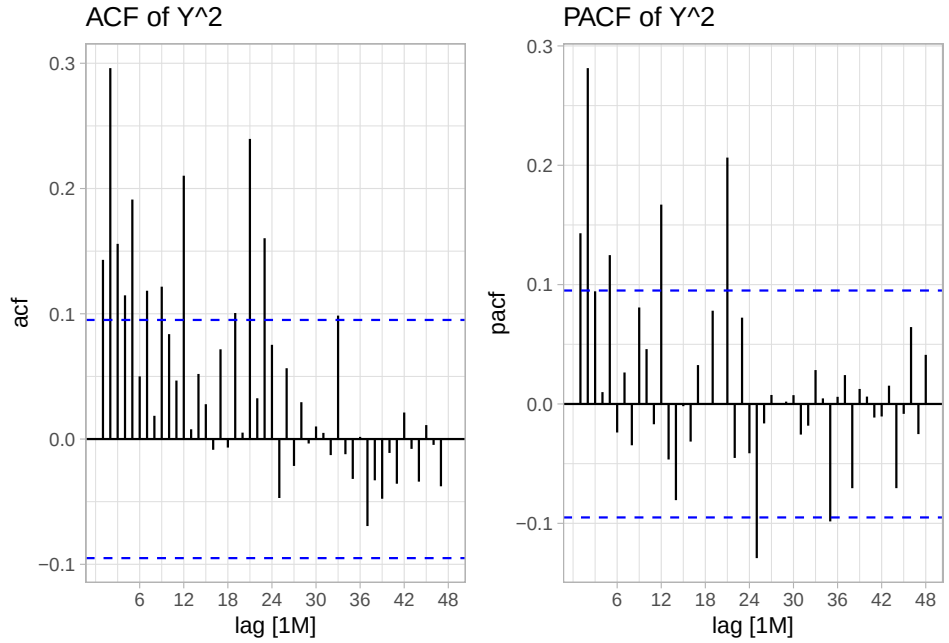
Instead, the second-place RMSE and the model with the lowest MAE are both the combined forecast. Thus, we will choose the combined model as our model of choice for when forecasting changes in California SNAP recipients.

(l) GARCH Model

For completeness, we will construct a model which accounts for heteroskedasticity, i.e. a generalized AutoRegressive Conditional Heteroskedasticity (GARCH) model.

ACF and PACF Plots

We have already seen and commented on the ACF and PACF plots for the time series earlier, but to assess possible ARCH/GARCH models, we will look at the ACF and PACF of the squared time series.



Looking at the ACF and PACF plots above, it is difficult to determine an immediate order of an ARCH model; if we were to chose, we might go with an ARCH(25). The high order of the proposed ARCH model indicates the likely need for a GARCH model instead. When we build our model, we will optimize over many combinations of p and q for GARCH(p,q).

Model Construction

Using`auto.arima()` we found the optimal ARMA model to be ARMA(1,1). Optimizing based on AIC, we then found the optimal GARCH order to be GARCH(1,1). Thus this model takes the following form:

$$y_t = \phi y_{t-1} + \theta \varepsilon_{t-1} + \sigma_{t|t-1} z_t$$

where

$$\sigma^2_{t|t-1} = \omega + \alpha \varepsilon^2_{t-1} + \beta \sigma^2_{t-1|t-2}$$

The parameter estimates are summarized in the table below. We observe that $\beta = 0.8545$, meaning that approximately 85% of the previous month's variance carries over into the current variance.

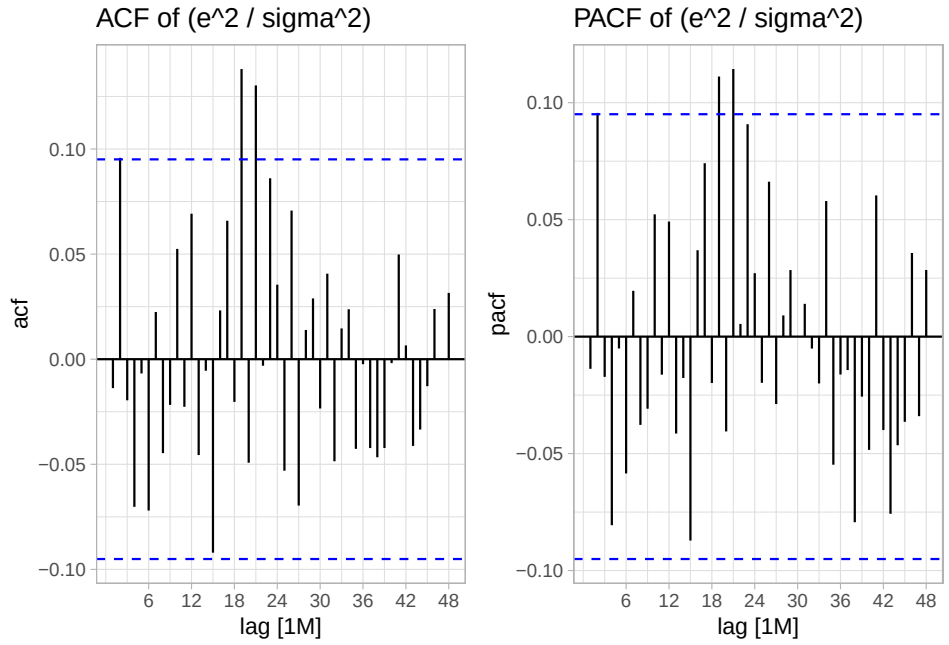
Table 8: GARCH Parameter Estimates

	AR(1)	MA(1)	Omega	Alpha	Beta	Skew	Shape
Estimates	0.9791	-0.7966	15.973	0.1209	0.8545	0.9976	11.6022

Also worth mentioning is that persistence was computed to be 0.8312701, so there is a fairly high persistence to the GARCH process.

Verifying GARCH Model

To see the effectiveness of our GARCH model, we have the ACF and PACF of $\hat{\varepsilon}_t^2/\hat{\sigma}_{t|t-1}^2$. Ideally, this plot should show no significant lags. We end up seeing statistical significance at lags 19 and 21, indicating possible issues.



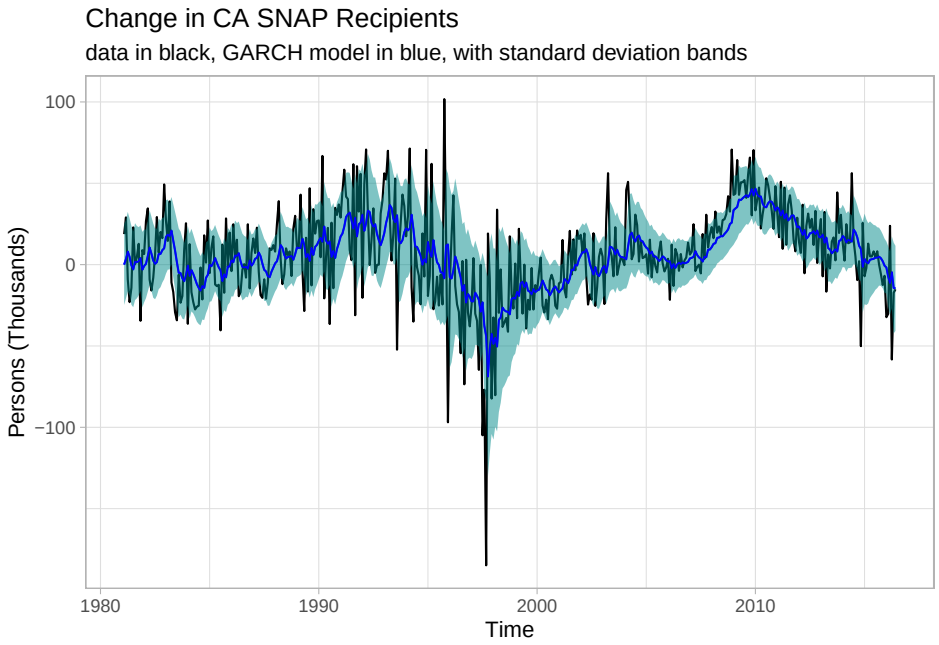
To further analyze serial correlation in the above plots, we performed two Ljung-Box tests. The first showed statistical significance using 24 lags (2 years), so serial correlation exists in that lag-window. The second showed no statistical significance using 48 lags (4 years), so in the long-term, the GARCH model was successful at eliminating serial correlation in $\hat{\varepsilon}_t^2/\hat{\sigma}_{t|t-1}^2$.

```
##
##  Box-Ljung test
##
## data:  GARCH_resid_sigma
## X-squared = 41.583, df = 24, p-value = 0.01436
```

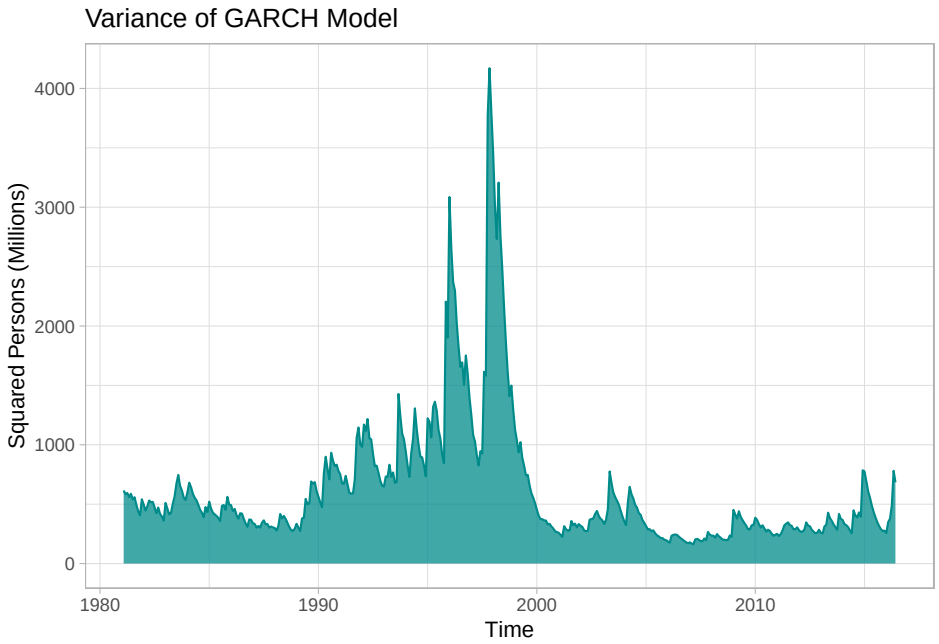
```
##
##  Box-Ljung test
##
## data:  GARCH_resid_sigma
## X-squared = 57.195, df = 48, p-value = 0.1707
```

GARCH Model Visualizations

To illustrate the fit of the model, the following graph shows the original time series (black curve), the model (blue curve), and standard deviation bands.



We can look at just the variance alone to see how it changes over time.

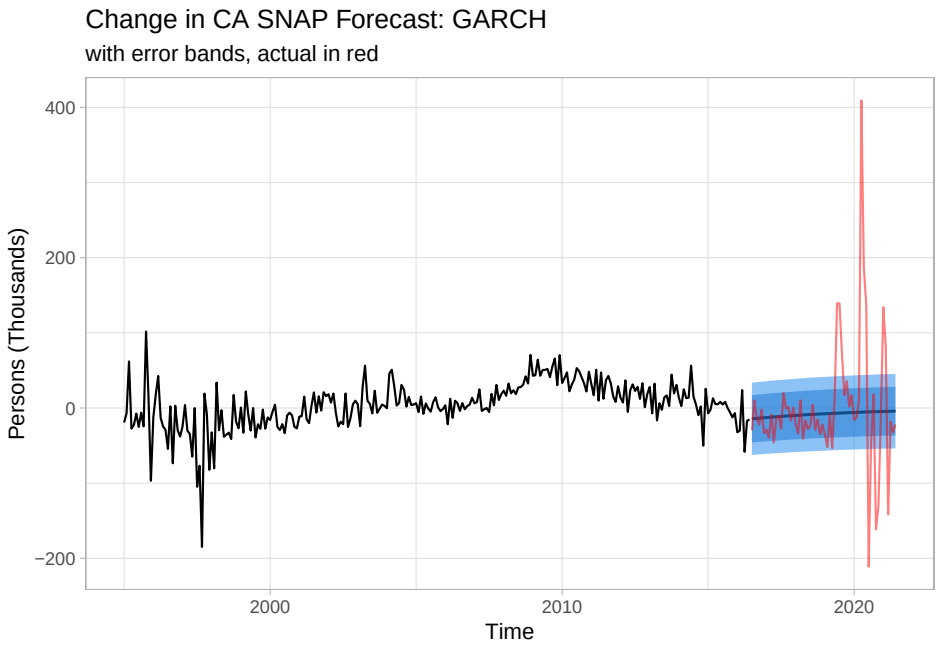


From the above plot, we see that variance tends to stay below 1000. There were, however, high spikes during the 1990s, indicating periods of high volatility.

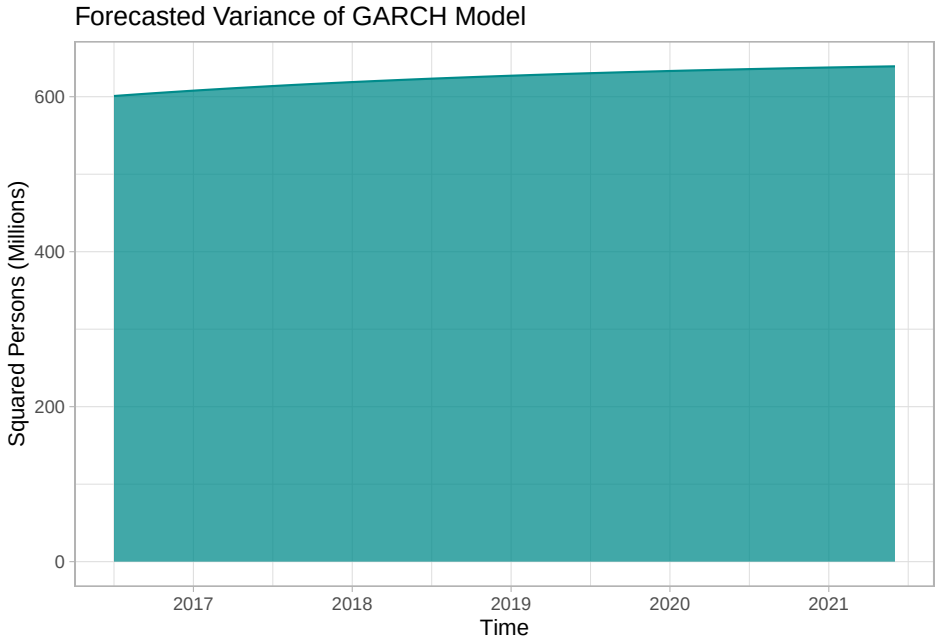
Forecast

Below is our 5-year forecast using the GARCH model. Since our GARCH model is based on an ARMA model, the forecast looks very similar to an ARMA forecast, with the relatively quick convergence to the mean. Like our other forecasts, the irregular behavior in the data starting in 2019 leads to issues in the model. For the error bands, we assumed that each forecast is conditionally normally distributed, with the

variance being the conditional variance modeled with the GARCH model.



In the below plot we graphed the forecasted conditional variance for the 5-year forecast. We observe the conditional variance increasing slightly, but not changing drastically. Clearly our forecast did not account for a large divergence from the trend, as we saw with the point forecast above.



Lastly, we compute the goodness-of-fit measures on the forecast. The MAE and RMSE are both high at 47.20175 and 84.5126. The mean error of 7.434645 indicates that the GARCH model underestimated the true behavior, in general.

```
## --- Testing Dataset ---
## MAE:  47.20175
## RMSE: 84.5126
## ME:   7.434645
```

III. Conclusion and Future Work

In summary, we attempted to fit 10 different models to changes in California SNAP recipients. The models were a model for trend, seasonality, and cycles, captured with an STL decomposition and an ARMA model for the cycles; an ARIMA model; an ETS model; a Holt-Winters seasonal model with additive seasonality; a TBATS model; a VAR model, with Washington SNAP recipients as the causal exogenous variable; a Prophet model; a neural network autoregression (NNETAR) model; a combination of the 8 above models using static weights; and a GARCH model.

We chose to not include the GARCH model in our combined model since its ability to capture trend is weaker than an ARIMA model, and we already have an ARIMA model in our set of models. Furthermore, the purpose of a GARCH model is to ultimately model conditional variance, so it falls in a separate category than all our other models.

To assess model performance, we performed a train-test split on the data, leaving 5 years (60 observations, July 2016 to June 2021) to be forecasted by the models. We evaluated the forecast performance using MAE and RMSE on the test data set. We found that the combined forecast performed the best in terms of MAE, and the VAR model perfomed the best in terms of RMSE. Interestingly, the VAR model performed the worst in terms of MAE, yet the combined forecast had the second best RMSE. Since the combined model performs comparatively well for both measures, we find that the combined model had the best overall performance.

We noticed that the forecast accuracy measures – MAE and RMSE – were very high in the context of the data. For example, the standard deviation of the data set was around 30, and yet the MAE for the models was in the 40s to 50s range. Even worse, the RMSE measures were in the 80s. Those high magnitudes indicate to us that none of the models performed exceedly well: they all struggled to forecast the testing data set. We also note that, in looking at the ME for each model on the testing data set, only the VAR model overestimated the data. All other models tended to underestimate the data.

Ultimately, there are two large potential sources of error when we look at our forecast performances on the test data set. First, there is the issue of increased volatility in the actual data. We noticed that all the forecasts seemed to capture the period up to 2019 very well, but at 2019 an onward, the data diverged significantly from our forecasts. We can analyze this reason given the context of the data: SNAP benefits are typically given to people in low-wage jobs to help get food and other resources. During 2019 and onward, the COVID-19 pandemic caused many people to lose jobs and many others to face food insecurity. In that time, demand for social welfare programs such as SNAP likely increased dramatically due to those changes to the job market and food distribution. To make things worse, our data was the *changes* in SNAP recipients, so monthly fluctuations were magnified during the differencing process. Since our models could not have predicted the COVID-19 pandemic and its effects, it is understandable that our forecasts performed poorly for those periods of irregular behavior.

The other potential source of error is overfitting. Even though 5 years (60 observations) is a long period of time to be forecasting, it is only 12.5% (or 1/8) of the total sample size. Typically we would prefer at least 20% to 30% of the data to be used in the test data set. Therefore, we might be able to attribute some of the large goodness-of-fit measures to overfitting, though the larger culprit is the irregular behavior of the data.

Future work in studying SNAP recipients would be very beneficial, especially in the domain of state-wide and national policy decisions, as well as the in the study of the effectiveness of government social welfare programs. Work in this field could lead to insight into poverty in the United States as well as a more informed view of government resource allocation.

IV. References

Department of Social Services, "CalFresh." *CA.gov*, 2023. https://www.cdss.ca.gov/calfresh.

Department of Social Services, "Eligibility and Issuance Requirements." *CA.gov*, 2023. https://www.cdss.ca.gov/inforesources/cdss-programs/calfresh/eligibility-and-issuance-requirements.

Diebold, F.X. (2017), *Forecasting*, Department of Economics, University of Pennsylvania, http://www.ssc.upenn.edu/~fdiebold/Textbooks.html.

Federal Reserve Bank of St. Louis, "SNAP Benefits Recipients in California." *United States Federal Reserve System*, 2023. https://fred.stlouisfed.org/series/BRCA06M647NCEN.

Federal Reserve Bank of St. Louis, "SNAP Benefits Recipients in Washington." *United States Federal Reserve System*, 2023. https://fred.stlouisfed.org/series/BRWA53M647NCEN.

Food and Nutrition Service, "A Short History of SNAP." *US Department of Agriculture*, 2023. https://www.fns.usda.gov/snap/short-history-snap.

Food and Nutrition Service, "SNAP Data Tables." *US Department of Agriculture*, 2023. https://www.fns.usda.gov/pd/supplemental-nutrition-assistance-program-snap.

Food and Nutrition Service, "SNAP Eligibility." *US Department of Agriculture*, 2023. https://www.fns.usda.gov/snap/recipient/eligibility.

Food and Nutrition Service, "Supplemental Nutrition Assistance Program (SNAP)." *US Department of Agriculture*, 2023. https://www.fns.usda.gov/snap/supplemental-nutrition-assistance-program.

Hyndman Rob J., and George Athanasopoulos. *Forecasting: Principles and Practice, 3rd Edition.* Online: OTexts, 2021. https://otexts.com/fpp3/.