Multivariate Analysis of Undocuments Immigrants in the United States

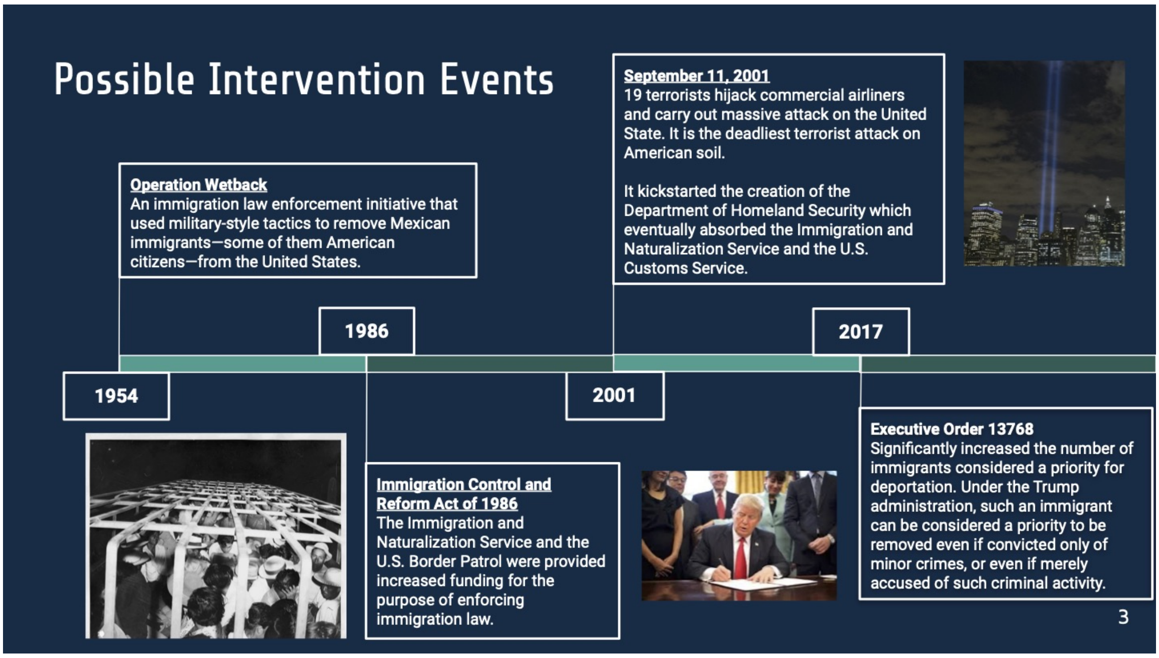
Harmit Chima, Amy Petris, Elizabeth Fabio

May 11, 2021

1 Background

The inspiration behind tracking a dataset from the 2019 Yearbook of Immigration Statistics (Department of Homeland Security) was from the Netflix documentary called "Immigration Nation". The documentary takes a deep look into recent U.S. immigration policies and its effect on undocumented immigrants and their families. The 2020 documentary highlights everything related to U.S. immigration, from the duties of an Immigration and Customs Enforcement (ICE) officer to the criminalization of undocumented immigrants.

The documentary also goes into detail significant government policies that had a huge effect on undocumented immigrants. Figure 1 gives a timeline of these major immigration events and policies.



Possible Intervention Events

Operation Wetback
An immigration law enforcement initiative that used military-style tactics to remove Mexican immigrants—some of them American citizens—from the United States.

September 11, 2001
19 terrorists hijack commercial airliners and carry out massive attack on the United State. It is the deadliest terrorist attack on American soil.

It kickstarted the creation of the Department of Homeland Security which eventually absorbed the Immigration and Naturalization Service and the U.S. Customs Service.

1954

1986

2001

2017

Immigration Control and Reform Act of 1986
The Immigration and Naturalization Service and the U.S. Border Patrol were provided increased funding for the purpose of enforcing immigration law.

Executive Order 13768
Significantly increased the number of immigrants considered a priority for deportation. Under the Trump administration, such an immigrant can be considered a priority to be removed even if convicted only of minor crimes, or even if merely accused of such criminal activity.

3

Figure 1: Major U.S. Immigration Policies

1.1 Data Source

The Yearbook of Immigration Statistics holds 41 tables that provides the number of immigrants in a variety of cases for each year. For example, some tables hold information regarding immigrants that were granted a green card or asylum/refugee status. Other tables, focused on this project, are related to immigration enforcement actions. These tables are:

- Table 33. Aliens Apprehended: Fiscal Years 1925 to 2019

- Table 39. Aliens Removed or Returned: Fiscal Years 1892 to 2019

Table 33 is the number of immigrants apprehended each year from ICE or the U.S. Border Patrol. As for Table 39, it lists the number of immigrants deported. This table has two columns: (1) immigrants deported based on an order of removal and (2) immigrants deported not based on an order of removal. The website was not clear regarding what was the difference between deportation based an order of removal and not from an order of removal. But from our understanding, an order of removal means that undocumented immigrants went through immigration court to make a plea. As for immigrants deported not due to an order of removal, this means that they voluntarily chose to be deported and to not go through the immigration court process.

After copying and pasting these tables into a CSV file, they were uploaded into Kaggle as public datasets:

- Undocumented Immigrants Apprehended in the U.S.
- Undocumented Immigrants Deported in the U.S.

After loading the two CSV files in R, we defined the datasets to the following variables:

- **num**: Number of Undocumented Immigrants Apprehended
- **rem**: Number of Undocumented Immigrants Deported Based on a Order of Removal
- **ret**: Number of Undocumented Immigrants Deported Not Based on a Order of Removal

1.2 Statistics

Table 1: Basic Summary Statistics

	Mean	Median	Standart Deviation
Apprehended	614, 880.9	530, 250	543, 896
Removed	67, 906.48	19, 312	112, 950.2
Returned	521, 888.5	30, 3348	501, 417.5

1.3 Presidential Analysis

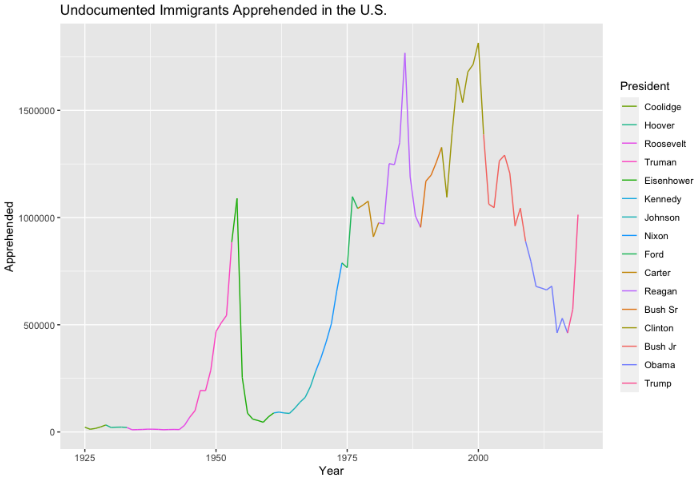


Figure 2: Time Series Plot of Apprehended by President

The 42nd president, Bill Clinton, had the highest number of undocumented immigrants apprehended during his terms with an average of 1, 526, 405.4 immigrants apprehended each year from 1993 to 2001. The 32nd president, Franklin D. Roosevelt, had the lowest average number of immigrants apprehended with approximately 13989.4 immigrants apprehended each year between 1933 and 1945.

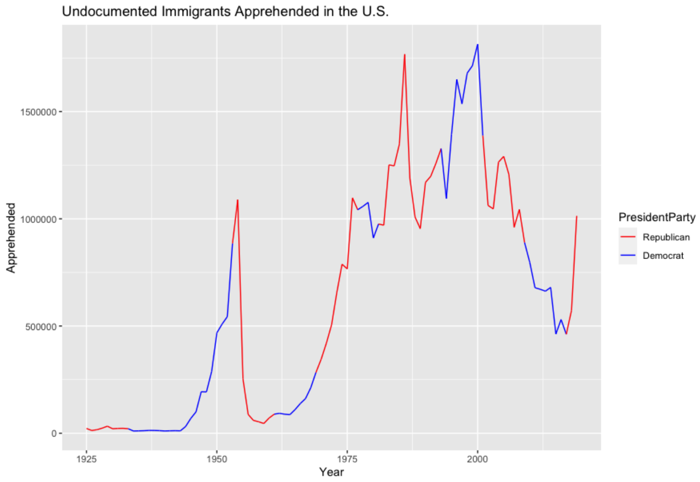


Figure 3: Time Series Plot of Apprehended by President Party

On average, 707,084 immigrants are apprehended under a Republican president, while 524,599 are apprehended under a Democratic president.

2 Model Building

For determining the model order, the following steps need to be taken:

1. **Check Stationarity**: You can check a dataset is stationary either by analyzing visualizations or through hypothesis testing. The visualizations to refer to are time series plots, autocorrelation function (ACF) plots, and partial autocorrelation function (PACF) plots. As for hypothesis testing, we can perform the augmented Dickey-Fuller unit-root test and the Box Cox transformation function to determine stationarity.

2. **Perform Data Transformation**: If a dataset is not stationary, the dataset needs to be applied with a log transformation and/or differenced. If the data has a non-constant mean, then we only need to take the difference. If the data has a non-constant variance, then the data needs to be applied with a log transformation. However, if the data has both a non-constant mean and a non-constant variance, then a log transformation needs to be applied first and then we can take the difference.

3. **Determine a Set of Preliminary Models**: Based on the ACF and PACF we can determine the autoregressive (AR) or the moving average (MA) order based on where the lag cuts off. We can also use the extended autocorrelation function (EACF) to determine the ARMA model as well.

4. **Select the Best Model Order**: Calculate the Akaike information criterion (AIC) for each model in question. The AIC is a measurement of how well the data fits the model. Essentially the best fit model has the smallest AIC value. Thus, the model with the lowest AIC is selected.

5. **Determine Coefficient Significance**: Apply the model into an ARIMA function in R and use the coeftest function to gain inference on the estimated coefficients. Create a new ARIMA model with only the significant coefficients and compare it with the original model. Select the model with the lowest AIC value.

2.1 Determine Model Order

Each model order will be determined for the three datasets (num, ret, rem).

2.1.1 Check Stationarity

First, we plot the time series and note any trends (Figures 4 & 5).

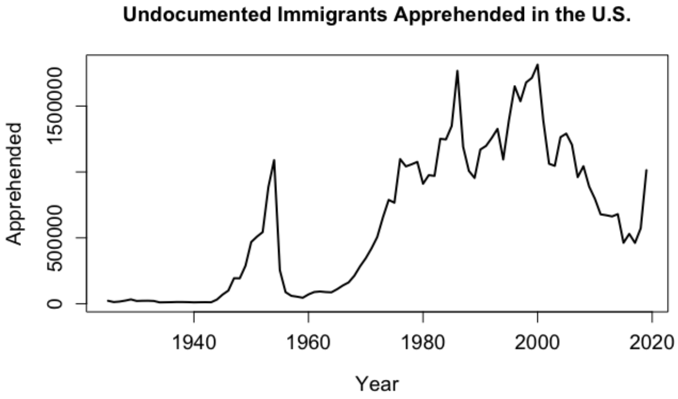


Figure 4: Time Series Plot of num

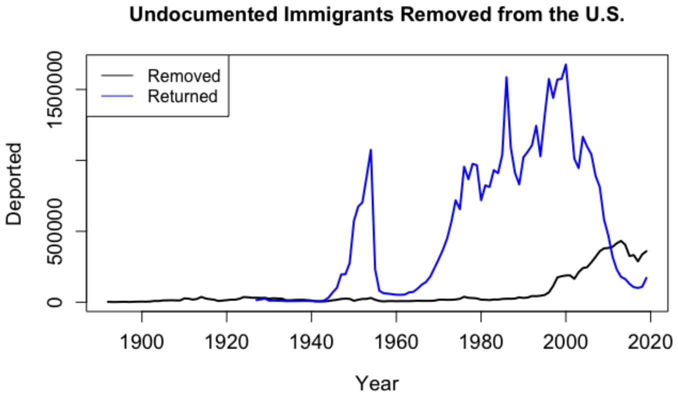


Figure 5: Time Series Plot of rem and ret

Both plots appear to have a non-constant variance considering there are several spikes especially around 1950 and near the early 2000s. As for non-constant variance, the datasets does not appear to need a log transformation. However, further tests needs to be done to confirm.

Next, we plot the ACF and PACF for each of the three datasets (Figures 6, 7, & 8). The ACF and PACF of all three datasets appear to have the lag tail off in the ACF plot while the PACF plot cuts off at lag 1. This shows that the models are currently AR. However, it is still hard to determine if the models are stationary just by those plots alone. Further proof from getting the roots of these AR models would be necessary.

Compared to using the visualization plots, we can use a variety of tests and functions to confirm the data's stationarity. For instance, using the BoxCox.lambda function in R predicts the lambda value. If lambda is close to zero, then a log transformation is needed. After applying the function to all three datasets, we have the following lambda values listed in Table 2.

To determine if the data needs to be differenced, we can use the augmented Dickey-Fuller unit-root test. The following conditions for this test are:

- $\underline{H_0}$: ARIMA(p, d, q) process with $d \neq 0$ (Data is not stationary).
- $\underline{H_a}$: ARIMA(p, d, q) process with $d = 0$ (Data is stationary).

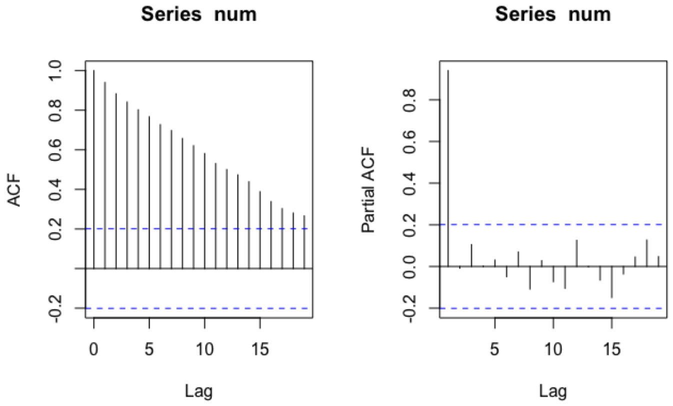


Figure 6: ACF & PACF of num

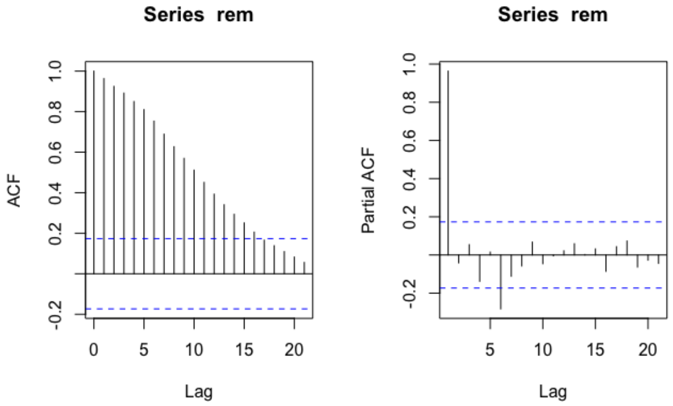


Figure 7: ACF & PACF of rem

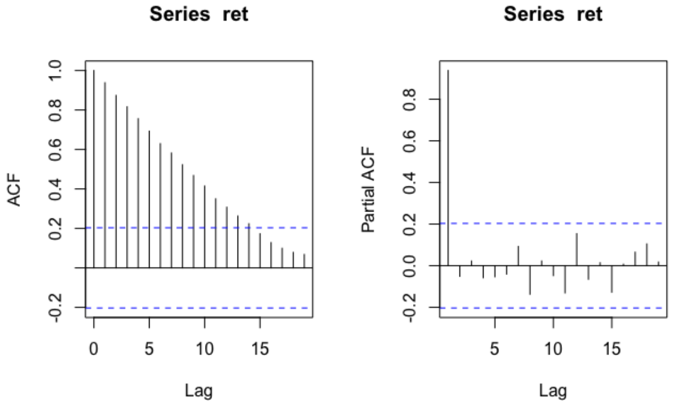


Figure 8: ACF & PACF of ret

Table 2: BoxCox.lambda Function Lambda Values

Dataset	Lambda	Log Transformation Needed?
num	-0.139	Yes
rem	0.172	Yes
ret	-0.042	Yes

- Condition: Reject H_0 if p-value is less than 0.05. This means if the p-value is greater than 0.05, the data is not stationary and needs to be differenced.

For the results from the augmented Dickey-Fuller unit root test for each dataset refer to Table 3.

Table 3: Augmented Dickey-Fuller Unit-Root Test p-Values

Dataset	p-Value	Difference Needed?
num	0.2082	Yes
rem	0.7145	Yes
ret	0.3364	Yes

2.1.2 Perform Data Transformation

After applying the log transformation first and then differencing the data, we have the following time series plots (Figure 9 & 10) and ACF and PACF plots (Figure 11, 12, & 13) for our three datasets.

The following variables represent the log transformed and differenced dataset:

- **dlognum**: Transformed num Dataset
- **dlogrem**: Transformed rem Dataset
- **dlogret**: Transformed ret Dataset

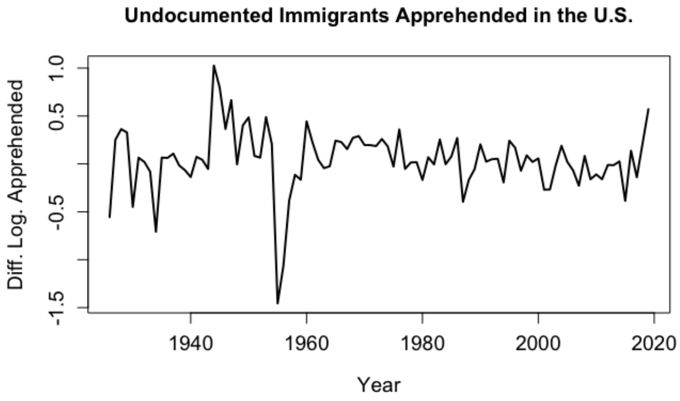


Figure 9: Time Series Plot of dlognum

Minus the outliers, our time series plots appear to be stationary along with the ACF and PACF plots. The time series plots do not have a general trend and the ACF and PACF plots have lags that are not not significant (within the dotted blue line).

To further confirm the newly transformed datasets are stationary, we performed the augmented Dickey-Fuller unit root test. The results are listed in Table 4.

Table 4: Augmented Dickey-Fuller Unit-Root Test p-Values on Transformed Data

Dataset	p-Value	Difference Needed?
dlognum	<0.01	No
dlogrem	<0.01	No
dlogret	0.04066	No

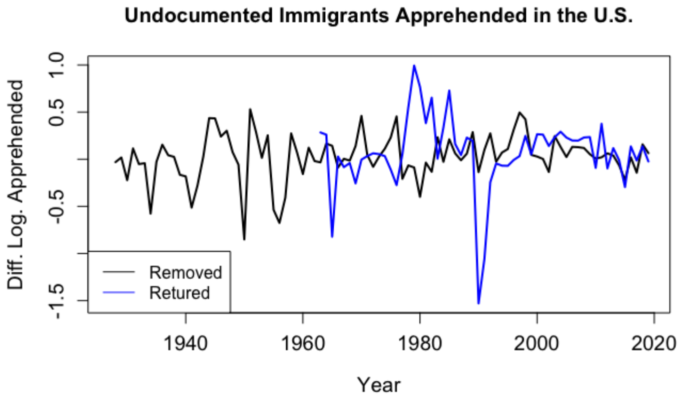


Figure 10: Time Series Plot of dlogrem and dlogret

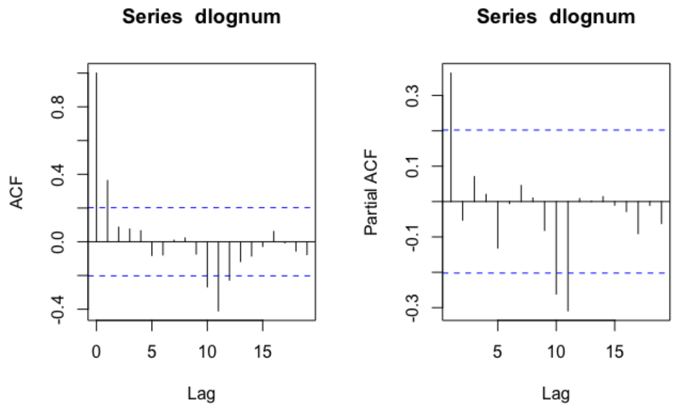


Figure 11: ACF & PACF of dlognum

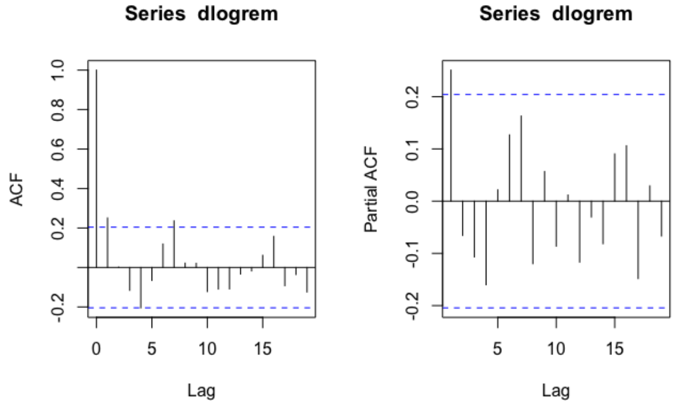


Figure 12: ACF & PACF of dlogrem

2.1.3 Determine a Set of Preliminary Models

We can refer to the ACF and PACF plots in Figures 11, 12, and 13 to make an inference regarding their model orders. For easier interpretation we used the EACF plot as well. To determine a model order on an EACF plot, we traced over a triangle of 0's that line the border. The better model is where the triangle index is closest to the top-left corner. In Figure 14, 15, and 16, the 0's highlighted in blue are the borders of the triangle.

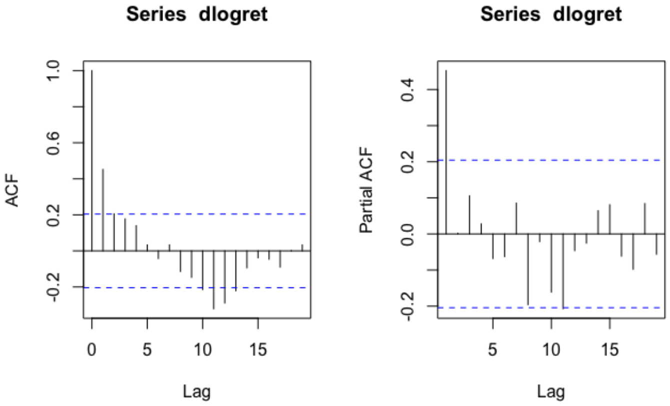
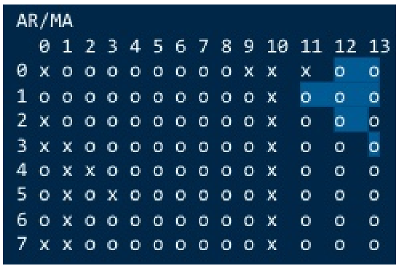


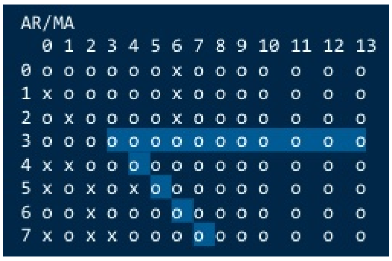
Figure 13: ACF & PACF of dlogret

Two triangles were selected in Figure 14. One triangle is where the index is at MA(12) and another is where the index is at ARMA(1,11). The reason why two triangles were selected instead of one is because MA(12) and ARMA(1,11) are close together. It would be hard to determine which model is the best unless its tested further.



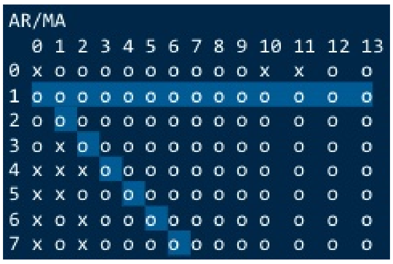
```
AR/MA
  0 1 2 3 4 5 6 7 8 9 10 11 12 13
0 x o o o o o o o o x x  x  o  o
1 o o o o o o o o o o x  o  o  o
2 x o o o o o o o o o x  o  o  o
3 x x o o o o o o o o x  o  o  o
4 o x x o o o o o o o x  o  o  o
5 o x o x o o o o o o x  o  o  o
6 o x o o o o o o o o x  o  o  o
7 x x o o o o o o o o x  o  o  o
```

Figure 14: EACF of dlognum



```
AR/MA
  0 1 2 3 4 5 6 7 8 9 10 11 12 13
0 o o o o o o o x o o o  o  o  o
1 x o o o o o x o o o o  o  o  o
2 o x o o o o x o o o o  o  o  o
3 o o o o o o o o o o o  o  o  o
4 x x o o o o o o o o o  o  o  o
5 x o x o x o o o o o o  o  o  o
6 o o x o o o o o o o o  o  o  o
7 x o x x o o o o o o o  o  o  o
```

Figure 15: EACF of dlogrem



```
AR/MA
  0 1 2 3 4 5 6 7 8 9 10 11 12 13
0 x o o o o o o o o o x  x  o  o
1 o o o o o o o o o o o  o  o  o
2 o o o o o o o o o o o  o  o  o
3 o x o o o o o o o o o  o  o  o
4 x x x o o o o o o o o  o  o  o
5 x x o o o o o o o o o  o  o  o
6 x o x o o o o o o o o  o  o  o
7 x o x o o o o o o o o  o  o  o
```

Figure 16: EACF of dlogret

So based on our EACF plots the model orders to test are:

- **num**: MA(12) and ARMA(1,11)
- **rem**: ARMA(3,3)
- **ret**: AR(1)

2.1.4 Select the Best Model Order

To select the best model order, an ARIMA model is created in R. We then compare the AIC values for each model and select the model order with the smallest AIC value. Since only the num dataset has two possible models, we only created the ARIMA models for that dataset and calculated the AIC values listed in Table 5.

Table 5: AIC Model Selection of dlognum

Model Order	AIC	Model Selected?
MA(12)	47.24	No
ARMA(1,11)	46.53	Yes

2.1.5 Determine Coefficient Significance

Next, we can compare the AIC values of ARIMA models that have only their significant coefficients. We first begin with using the coeftest function in R to determine each significant coefficient within the 95% confidence interval. Next, we created another ARIMA model with the same order but only having their significant variables selected. Then we compared the two similar models and chose the one with the best AIC value.

In Table 6, we listed each possible model along with its AIC values. For the num dataset, we checked again MA(12) with only the significant coefficients even though ARMA(1,11) was originally selected.

Table 6: Final AIC Model Selection

Dataset	Model Order	Significant Coefficients Only?	AIC	Model Selected?
num	ARMA(1,11)	No	44.53	No
num	ARMA(1,11)	Yes	38.00	No
num	MA(12)	No	45.24	No
num	MA(12)	Yes	32.97	Yes
rem	ARMA(3,3)	No	26.75	Yes
rem	ARMA(3,3)	Yes	30.06	No
ret	AR(1)	No	41.05	No
ret	AR(1)	Yes	39.38	Yes

2.2 Model Diagnostics

Before using the selected model order for further analysis, we need to perform the following diagnostic steps:

1. Step 1: Perform residual analysis to check for underspecification.
2. Step 2: Check for overspecification. This is performed in the last step already when determining the model order (Section 2.1.5).
3. Step 3: Check for stationarity through the AR roots.
4. Step 4: Check for model redundancy through the AR and MA roots.

2.2.1 Residual Analysis - Underspecification

For the residual analysis, the time series and the ACF and PACF were plotted first. The time series residual plots can be seen in Figure 17, 18, and 19, and the ACF and PACF residual plots can be seen in Figure 20, 21, and 22.

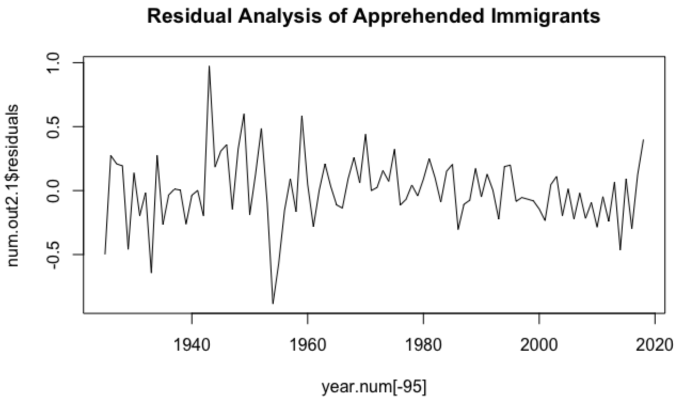


Figure 17: Residual Time Series Plot of num MA(12) Model with Significant Coefficients Only

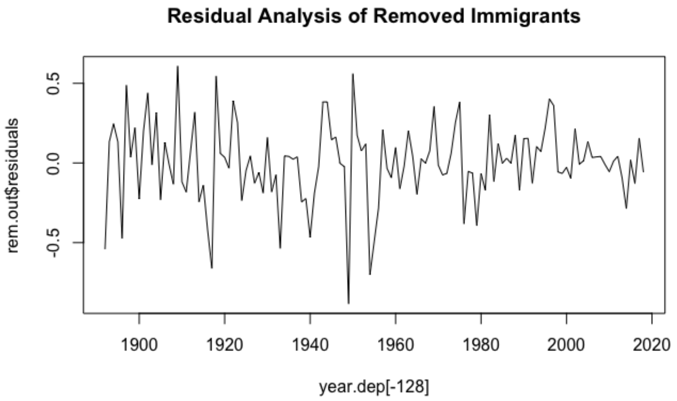


Figure 18: Residual Time Series Plot of rem ARMA(3,3) Model

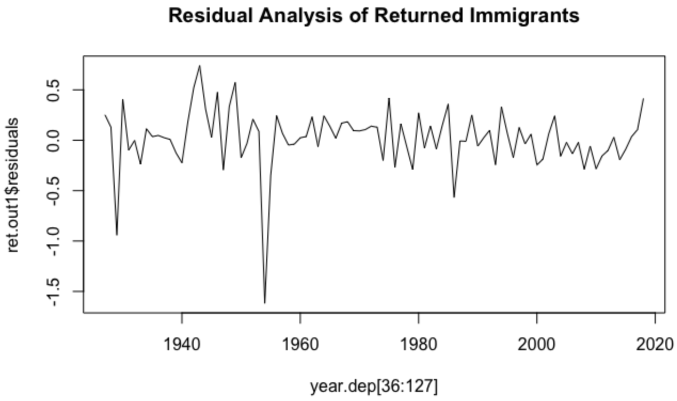


Figure 19: Residual Time Series Plot of ret AR(1,1) Model with Significant Coefficients Only

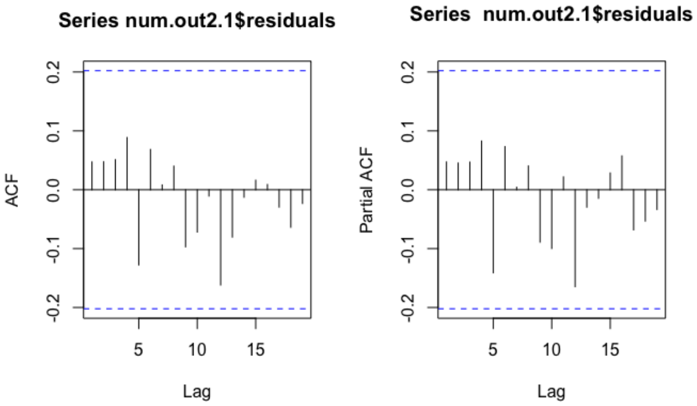


Figure 20: Residual ACF & PACF of num MA(12) With Significant Coefficients Only

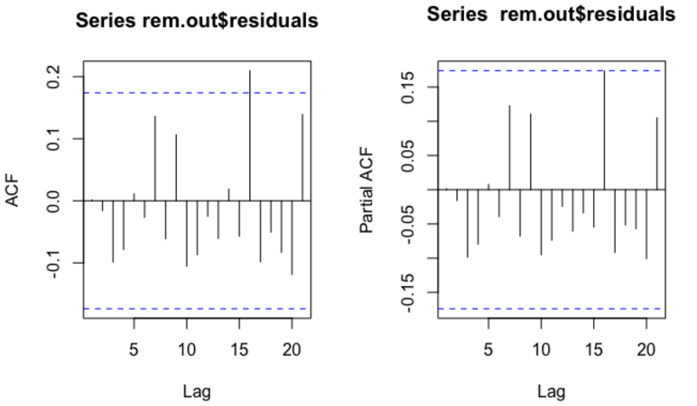


Figure 21: Residual ACF & PACF of rem ARMA(3,3) Model

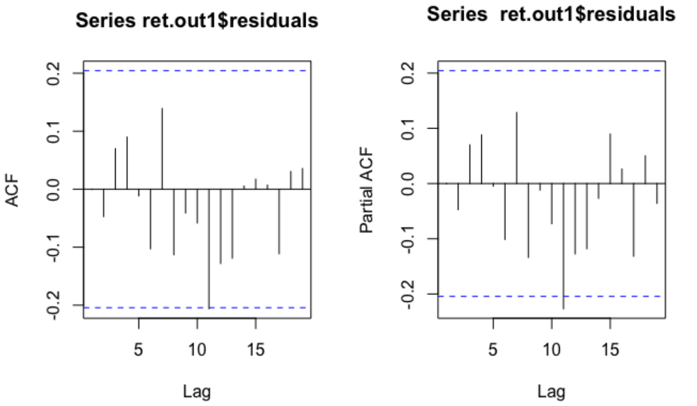


Figure 22: Residual ACF & PACF of ret AR(1,1) Model with Significant Coefficients Only

The residual time series plots for all the datasets appears to be generally white noise as well as the ACF and PACF plots. To confirm if the residuals for the selected models are white noise, we can perform the Box-Ljung test. The conditions for this test are:

- $\underline{H_0}$: $\rho_1 = \rho_2 = ... = 0$ (white noise)
- $\underline{H_a}$: At least one $\rho_k \neq 0$ (not white noise)
- Condition: Reject H_0 if p-value is less than 0.05. This means if the p-value is greater than

0.05 the residuals are not white noise.

All the models' residuals are white noise based on their p-values listed in Table 7. Thus, our selected models are not underspecified.

Table 7: Box-Ljung Test p-Values

Dataset	Model	p-Value	Residual Are White Noise?
num	MA(12) With Significant Coefficients Only	0.7654	Yes
rem	ARMA(3,3)	0.6548	Yes
ret	AR(1) With Significant Coefficients Only	0.3927	Yes

2.2.2 Overspecfication

The coeftest output for each model order along with the model orders with only the significant coefficients is listed below. To see which model was selected, refer to Section 2.1.5.

Coeftest Output for MA(12) - num Dataset

```
z test of coefficients:

           Estimate Std. Error z value  Pr(>|z|)
ma1        0.341711   0.127292  2.6845 0.0072644 **
ma2       -0.010666   0.131395 -0.0812 0.9353046
ma3       -0.023095   0.164579 -0.1403 0.8884000
ma4        0.143448   0.143925  0.9967 0.3189148
ma5       -0.070559   0.159197 -0.4432 0.6576104
ma6       -0.413222   0.159013 -2.5987 0.0093585 **
ma7       -0.091808   0.123955 -0.7407 0.4589009
ma8       -0.020123   0.130787 -0.1539 0.8777163
ma9       -0.073581   0.151119 -0.4869 0.6263233
ma10      -0.029191   0.127751 -0.2285 0.8192582
ma11      -0.572337   0.168295 -3.4008 0.0006719 ***
ma12      -0.180392   0.160995 -1.1205 0.2625080
intercept  0.049395   0.010147  4.8681 1.127e-06 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Coeftest Output for MA(12) Significant Coefficients Only - num Dataset

```
z test of coefficients:

           Estimate Std. Error z value  Pr(>|z|)
ma1        0.290715   0.087107  3.3374 0.0008456 ***
ma6       -0.318710   0.094855 -3.3600 0.0007795 ***
ma11      -0.525361   0.104283 -5.0379 4.708e-07 ***
intercept  0.047960   0.015363  3.1219 0.0017972 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Coeftest Output for ARMA(1,11) - num Dataset

```
z test of coefficients:

            Estimate Std. Error z value  Pr(>|z|)
ar1        0.3891286  0.2287320  1.7012  0.088897 .
ma1       -0.0271785  0.2517230 -0.1080  0.914020
ma2       -0.1083195  0.1424819 -0.7602  0.447115
ma3        0.0019829  0.2221487  0.0089  0.992878
```

```
ma4        0.1297223  0.1292946  1.0033  0.315712
ma5       -0.1499183  0.2098099 -0.7145  0.474891
ma6       -0.3870514  0.1936903 -1.9983  0.045684 *
ma7        0.0935913  0.1742246  0.5372  0.591138
ma8        0.0225764  0.1723225  0.1310  0.895765
ma9       -0.0529913  0.1550539 -0.3418  0.732531
ma10       0.0444361  0.1372179  0.3238  0.746062
ma11      -0.5668366  0.1751264 -3.2367  0.001209 **
intercept  0.0490834  0.0109852  4.4681 7.891e-06 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Coeftest Output for ARMA(1,11) Significant Coefficients Only - num Dataset

```
z test of coefficients:

            Estimate Std. Error z value  Pr(>|z|)
ma6       -0.4449770  0.1313688 -3.3872  0.000706 ***
ma11      -0.6056635  0.1296412 -4.6718 2.985e-06 ***
intercept  0.0492252  0.0082355  5.9772 2.270e-09 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Coeftest Output for ARMA(3,3) - rem Dataset

```
z test of coefficients:

           Estimate Std. Error  z value Pr(>|z|)
ar1       -0.227200   0.344147  -0.6602   0.5091
ar2       -0.982243   0.069896 -14.0530   <2e-16 ***
ar3       -0.024220   0.340169  -0.0712   0.9432
ma1        0.388339   0.331167   1.1726   0.2409
ma2        1.034128   0.058114  17.7947   <2e-16 ***
ma3        0.253686   0.334868   0.7576   0.4487
intercept  0.038244   0.026560   1.4399   0.1499
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Coeftest Output for ARMA(3,3) Significant Coefficients Only - rem Dataset

```
z test of coefficients:

    Estimate Std. Error z value Pr(>|z|)
ar2  0.34384    1.44804  0.2374   0.8123
ma2 -0.42053    1.40708 -0.2989   0.7650
```

For this model even though the AIC was higher versus its standard ARMA(3,3) model, it would have not been selected anyways since the coefficients are not significant.

Coeftest Output for AR(1) - ret Dataset

```
z test of coefficients:

          Estimate Std. Error z value  Pr(>|z|)
ar1       0.458533   0.093028  4.9290 8.266e-07 ***
intercept 0.032649   0.056439  0.5785    0.5629
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Coeftest Output for AR(1) Significant Coefficients Only - rem Dataset

```
z test of coefficients:

    Estimate Std. Error z value  Pr(>|z|)
ar1  0.46159    0.09304  4.9612 7.006e-07 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

2.2.3 Stationarity Using AR Roots

Another way to check the stationarity of an ARIMA model is by looking at the root of the AR component. If one of the roots is close to one, then differencing should be considered again.

Since the num dataset has a MA(12) model order, it is automatically stationary. So checking the AR root is not necessary considering it does not have an AR component.

As for the rem dataset, the AR roots of the model is listed below:

```
[1]  1.011585  1.011585 40.348045
```

Since some of the values are close to one, we differenced the model again with the model order ARIMA(3,1,3). We also created the same ARIMA model order with only the significant variables. After comparing the AIC values (Table 8), the original ARMA(3,3) is still the better fit.

Table 8: AIC Model Selection of dlogrem

Model Order	AIC	Model Selected?
ARMA(3,3)	26.75	Yes
ARIMA(3,1,3)	36.03	No
ARIMA(3,1,3) With Significant Coefficients Only	89.60	No

Lastly, the AR roots of the ret AR(1) model with significant coefficients only is definitely greater than one:

```
[1] 2.166429
```

Thus, this selected model is stationary.

2.2.4 Model Redundancy

To check for model redundancy, we look into not only the AR roots but also the MA roots. If the roots are almost equal to each other, then we would reduce the AR and MA order down one.

Since the selected num dataset model has a MA(12) order and the selected ret dataset model has an AR(1) order, model redundancy is not an issue. However, the rem dataset model has an ARMA(3,3) order that needs to be checked out.

The AR root of ARMA(3,3):

```
[1]  -0.103567+1.00627i  -0.103567-1.00627i -40.348045+0.00000i
```

The MA root of ARMA(3,3):

```
[1] -0.0673246+0.9977465i -0.0673246-0.9977465i -3.9417633+0.0000000i
```

The AR and MA roots do not match in this case. Thus there is no model redundancy on the ARMA(3,3) model for the rem dataset.

3 Forecasting

We predicted the last five years we had data for, these years include 2015 to 2019. We did this for the log of the number apprehended (lognum) and the number deported which were separated into the number returned (logret) and the number removed (logrem). Due to eliminating the last five years of each data set, new variables were created.

```
new_lognum = head(lognum,-5)
new_logret = head(logret,-5)
new_logrem = head(logrem,-5)
```

3.1 Number Apprehended

After adding back the difference order, we narrowed it down to two estimated models: ARMA(1,1,11) and ARMA(0,1,12).

```
num.out2.2 <- arima(new_lognum, order = c(1, 1, 11))

Call:
arima(x = new_lognum, order = c(1, 1, 11))

Coefficients:
         ar1     ma1      ma2     ma3     ma4      ma5      ma6     ma7     ma8      ma9    ma10
      0.3030  0.1588  -0.0428  0.0606  0.1746  -0.0191  -0.3057  0.1270  0.0824  -0.0113  0.1037
s.e.  0.3237  0.3308   0.1530  0.1442  0.1243   0.1770   0.1111  0.1643  0.1166   0.1174  0.1248
        ma11
     -0.4923
s.e.   0.1212

sigma^2 estimated as 0.06691:  log likelihood = -10.13,  aic = 44.25

num.out2.3 <- arima(new_lognum, order = c(0, 1, 12)

Call:
arima(x = new_lognum, order = c(0, 1, 12))

Coefficients:
         ma1     ma2     ma3     ma4     ma5      ma6     ma7     ma8     ma9    ma10     ma11
      0.4523  0.0755  0.0808  0.2033  0.0663  -0.2955  0.0252  0.1000  0.0120  0.0849  -0.4760
s.e.  0.1216  0.1203  0.1330  0.1257  0.1337   0.1186  0.1207  0.1158  0.1298  0.1283   0.1351
        ma12
     -0.0921
s.e.   0.1245

sigma^2 estimated as 0.06713:  log likelihood = -10.29,  aic = 44.58
```

Next, we check both estimated ARIMA models with only significant values.

```
Call:
arima(x = new_lognum, order = c(1, 1, 11), fixed = c(0, 0, 0, 0, 0, 0, NA, 0,
    0, 0, 0, NA))

Coefficients:
      ar1  ma1  ma2  ma3  ma4  ma5      ma6  ma7  ma8  ma9  ma10     ma11
        0    0    0    0    0    0  -0.1108    0    0    0     0  -0.3122
s.e.    0    0    0    0    0    0   0.0997    0    0    0     0   0.0854
```

```
sigma^2 estimated as 0.09165:  log likelihood = -20.55,  aic = 45.11

Call:
arima(x = new_lognum, order = c(0, 1, 12), fixed = c(NA, 0, 0, 0, 0, NA, 0,
    0, 0, 0, NA, 0))

Coefficients:
         ma1  ma2  ma3  ma4  ma5      ma6  ma7  ma8  ma9  ma10      ma11  ma12
      0.3946    0    0    0    0  -0.1593    0    0    0     0  -0.3588     0
s.e.  0.1022    0    0    0    0   0.1122    0    0    0     0   0.0990     0

sigma^2 estimated as 0.07694:  log likelihood = -13.3,  aic = 32.6
```

The estimated model order (0,1,12) had a smaller AIC value and was chosen with significant coefficients.

$\Delta_1 X_t = (1 + 0.3946B - 0.1593B^6 - 0.3588B^{11})\epsilon_t$

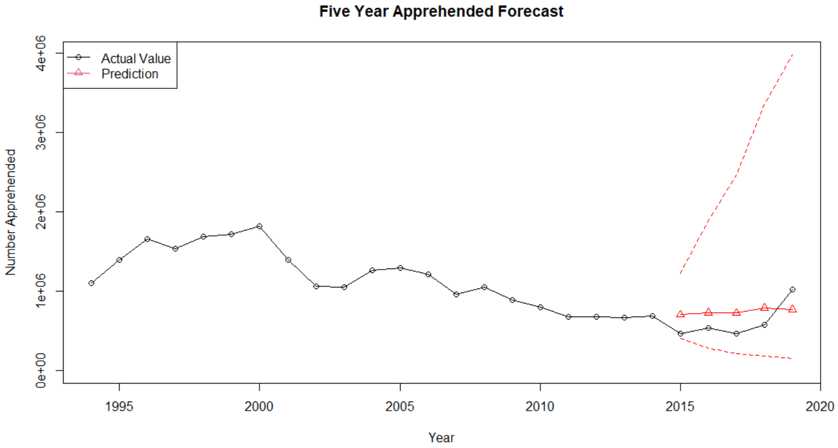


Figure 23: Forecast Plot of logapp ARMA(0,1,12) Model with significant coefficients

3.2 Number Removed

After adding back the difference order, the only model that was worth checking was ARIMA(3,1,3), which has an AIC value of -0.7.

```
Call:
arima(x = logrem, order = c(3, 1, 3))

Coefficients:
         ar1     ar2      ar3      ma1      ma2     ma3
      0.5625  0.0256  -0.7593  -0.3872  -0.2082  0.8570
s.e.  0.1445  0.1927   0.1396   0.1212   0.1517  0.1394

sigma^2 estimated as 0.05002:  log likelihood = 6.35,  aic = 1.3
```

These are the significant coefficients:

```
z test of coefficients:

Estimate Std. Error z value  Pr(>|z|)
```

```
ar1  0.562496   0.144501  3.8927 9.914e-05 ***
ar2  0.025558   0.192660  0.1327  0.894464
ar3 -0.759337   0.139646 -5.4376 5.400e-08 ***
ma1 -0.387202   0.121248 -3.1935  0.001406 **
ma2 -0.208241   0.151683 -1.3729  0.169792
ma3  0.857048   0.139368  6.1495 7.770e-10 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Since removing the non-significant coefficients increased the AIC value, no coefficients were removed.

$$(1+0.9399B_2-0.2035B_3)(1\breve{}B)1X_t=(1+0.1427B+B_2)\epsilon_t$$

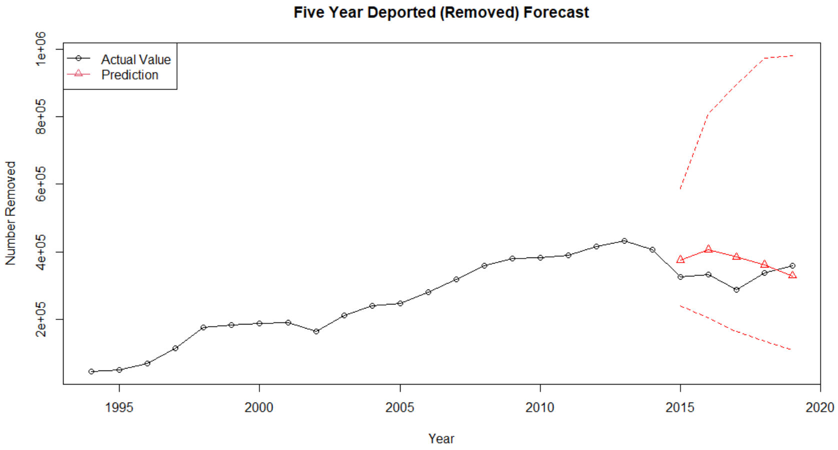


Figure 24: Forecast Plot of logrem ARMA(3,1,3) Model

3.3 Number Returned

After adding back the difference order, the only model to check was ARIMA(1,1,0).

```
Call:
arima(x = new_logret, order = c(1, 1, 0))

Coefficients:
         ar1
      0.4556
s.e.  0.0948

sigma^2 estimated as 0.09007:  log likelihood = -18.85,  aic = 39.7
>
```

This model had an AIC value of 39.7 with the the significant coefficients listed as:

```
z test of coefficients:

Estimate Std. Error z value  Pr(>|z|)
ar1 0.455631   0.094818  4.8053 1.545e-06 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

$$\hat{Y}_t=\mu+Y_(t-1)+0.4556(Y_(t-1)\breve{}Y_(t-2))$$

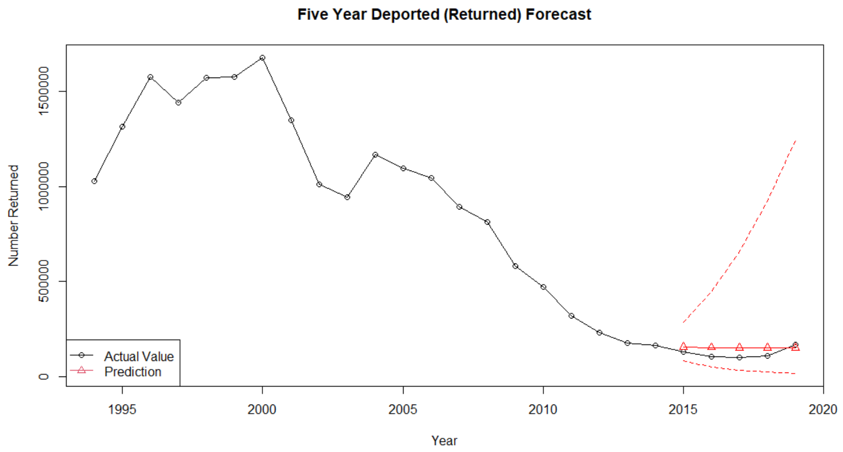


Figure 25: Forecast Plot of logret ARMA(1, 1, 0) Model

4 Intervention Events

Before analyzing interventions, we need to identify the significant outliers. We first perform the locate.outliers function in R on the residuals of our three models.

For our selected app model, no outliers were identified in the residual data. However, for the deported models (rem and ret), a few outliers were found:

Residual Outliers from Selected rem Model

```
  type ind    coefhat     tstat
1   AO  26 -0.6373329 -3.696818
2   AO  58 -0.8493135 -4.765517
3   TC  63 -0.5904907 -3.897957
```

Residual Outliers from Selected ret Model

```
  type ind    coefhat     tstat
1   AO   3 -0.9314625 -5.350058
2   AO  27  0.6879634  3.951468
5   TC  16  0.7165402  3.939461
6   TC  17  0.7910875  4.349314
7   TC  28 -1.4838573 -8.158088
```

Both datasets have different indexes since the ret dataset has less rows. However, at point 63 for the rem dataset and at point 28 for ret dataset, they both appear to be outliers for the year 1953.

4.1 Operation Wetback

Around 1953, the Bracero program was in full swing which allowed Mexican laborers to work in the U.S. for in exchange stricter border security and the return of illegal Mexican immigrants. However, the program did not work since more American growers continued to recruit illegal immigrants for work. A lot of Mexican laborers took the opportunity to move to the U.S. because of the food shortages at home.

Eventually many prominent Mexican farm owners were unhappy of the situation since they were losing potential workers to the U.S. So they applied pressure to the Mexican government to address the problem who went to the U.S. government to discuss solutions. One of these solutions was Operation Wetback.

Operation Wetback is an immigration enforcement initiative set in June 1954 that used military-style tactics to remove Mexican immigrants from the U.S. So it makes sense that the year before Operation Wetback was initiated, the the records show a spike in the number of removed and returned immigrants. Thus, for our intervention analysis we will look into the year 1953 for the rem and ret dataset.

4.2 Select the Best Intervention

The locate.outliers function recognize the outlier type at 1953 for the rem and ret dataset as "TC" or "Temporary Change". However, before we assume the intervention is only temporary, we can check by creating a variety of ARIMAX models with the following intervention conditions:

- None
- Temporary Intervention
- Permanent Intervention
- Temporary and Permanent Intervention

After the ARIMAX models are created, we can select the best one with the lowest AIC value. Table 9 lists each rem intervention along with its AIC value. The best model to proceed forward in plotting its total impact is the temporary and permanent intervention.

Table 9: Intervention Analysis on the rem Dataset

Intervention	AIC	Model Selected?
None	26.95	No
Temporary	28.57	No
Permanent	26.44	No
Temporary & Permanent	22.34	Yes

As for the ret dataset, we plot a similar table that lists the interventions and its AIC values (Table 10). However, the best model selected for this case is only the permanent intervention.

Table 10: Intervention Analysis on the ret Dataset

Intervention	AIC	Model Selected?
None	39.38	No
Temporary	20.82	No
Permanent	18.89	Yes
Temporary & Permanent	NaN	No

4.3 Total Impact

The total impact shows the effect of an outlier from that point on and out. Even though the 1953 outlier was a temporary change for both the rem and ret datsets (as reflected in the time series plot and the locate.outliers function), the intervention overall reflects a more permanent change. This can be seen in our total impact plots in Figure 26 and 27.

The intervention analysis overall is valid since it shows how Operation Wetback set a precedence in the following years regarding immigration policy. It may have not been an instant change, but an upward trend took shape around the 1980s when more immigration policies were created.

5 Multivariate Time Series Analysis

Since num, rem, and ret have no causal relationship, a multivariate time series analysis is considered over a time series regression. We can begin by creating a vector autoregresssion (VAR) model with the p order at one. A VAR model essentially extends the idea of a univariate AR

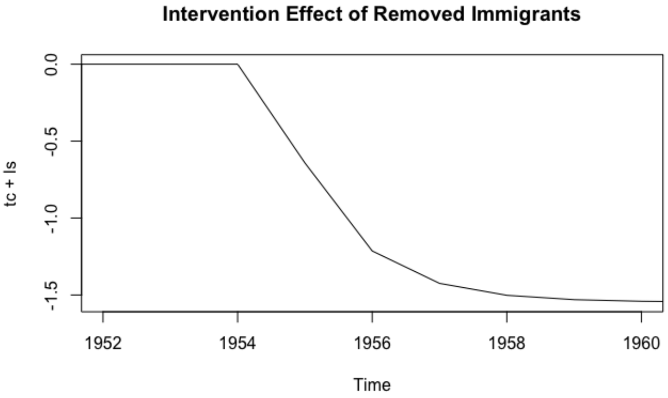


Figure 26: Total Impact Plot of rem

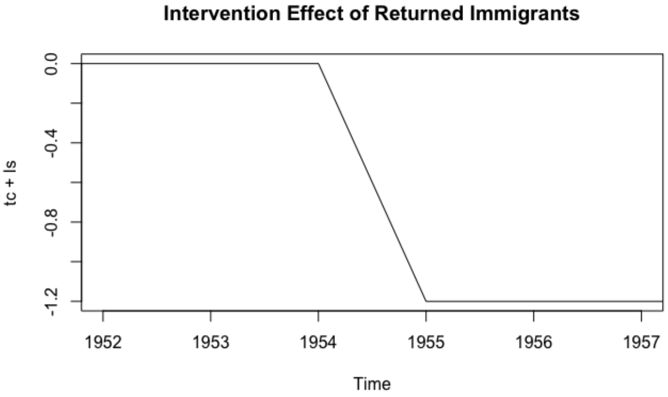


Figure 27: Total Impact Plot of ret

model to a multivariate time series regression.

To determine the best p order before creating a VAR model, we can call for the VARselect function in R. Based on the output, the optimum p order is one.

When calling the VAR function in R with $p = 1$, we get the following output:

```
VAR Estimation Results:
=========================
Endogenous variables: num, rem, ret
Deterministic variables: const
Sample size: 92
Log Likelihood: -3387.267
Roots of the characteristic polynomial:
0.9895 0.9672 0.9672
Call:
VAR(y = x, p = 1)


Estimation results for equation num:
====================================
num = num.l1 + rem.l1 + ret.l1 + const

         Estimate Std. Error t value Pr(>|t|)
num.l1     1.2524     0.2021   6.198  1.8e-08 ***
```

```
rem.l1    -0.3253     0.2213  -1.470   0.1452
ret.l1    -0.3183     0.2093  -1.521   0.1318
const  48449.0075 28250.4532   1.715   0.0899 .
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1


Residual standard error: 173400 on 88 degrees of freedom
Multiple R-Squared: 0.901,Adjusted R-squared: 0.8976
F-statistic: 266.8 on 3 and 88 DF,  p-value: < 2.2e-16


Estimation results for equation rem:
====================================
rem = num.l1 + rem.l1 + ret.l1 + const

         Estimate Std. Error t value Pr(>|t|)
num.l1 -5.107e-02  1.893e-02  -2.697  0.00839 **
rem.l1  1.052e+00  2.074e-02  50.697  < 2e-16 ***
ret.l1  6.216e-02  1.961e-02   3.170  0.00210 **
const  -1.640e+03  2.647e+03  -0.619  0.53724
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1


Residual standard error: 16240 on 88 degrees of freedom
Multiple R-Squared: 0.9843,Adjusted R-squared: 0.9837
F-statistic:  1836 on 3 and 88 DF,  p-value: < 2.2e-16


Estimation results for equation ret:
====================================
ret = num.l1 + rem.l1 + ret.l1 + const

         Estimate Std. Error t value Pr(>|t|)
num.l1     0.3402     0.1861   1.828  0.07087 .
rem.l1    -0.5258     0.2038  -2.580  0.01154 *
ret.l1     0.6047     0.1927   3.138  0.00231 **
const  42057.5345 26011.1872   1.617  0.10948
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1


Residual standard error: 159600 on 88 degrees of freedom
Multiple R-Squared: 0.902,Adjusted R-squared: 0.8986
F-statistic: 269.9 on 3 and 88 DF,  p-value: < 2.2e-16



Covariance matrix of residuals:
          num       rem       ret
num 3.005e+10 1.018e+09 2.569e+10
rem 1.018e+09 2.639e+08 5.937e+08
ret 2.569e+10 5.937e+08 2.548e+10

Correlation matrix of residuals:
       num    rem    ret
num 1.0000 0.3616 0.9286
```

```
rem 0.3616 1.0000 0.2290
ret 0.9286 0.2290 1.0000
```

Based on the first part of the VAR output regarding estimation results, the num variable only has one significant coefficient. That coefficient is itself. As for the rem dataset, all the variables have significant coefficients along with itself. The ret dataset though only has significant coefficients with the rem variable and itself.

For the covariance and correlation matrix section of the VAR(1) output, there appears to be a stronger relationship between the num and ret dataset. This makes sense considering their time series plot is similar in shape and range. As for rem dataset, it has some relationship with num and ret but it is not as strong.

5.1 Impulse Response Function

A similar inference from the covariance and correlation matrix can also be observed when plotting the impulse response function (IRF). The x-axis of the plot is standard error, or a unit-less shock value, and the y-axis lists the three datasets response values. The red dotted line is the 95% confidence interval and red solid horizontal line is where $y = 0$. Essentially, the IRF plot shows the response of each dataset when a certain number of shock is applied from one variable.

For instance, Figure 28 shows the response of the three variables when a num is applied. As the shock increases, the rem variable is not affected by it compared to the ret variable and itself where the response is decreasing.

As for the rem variable in Figure 29, there is not a strong response from any of the variables since the plot is close to 0 (solid red line).

In Figure 30, the ret IRF plot is similar to the num IRF plot. The ret variable decreases the response of itself and the num variable.

When analyzing the relationship between num and ret we can infer that if more immigrants are apprehended, then more immigrants are in the "system". This means undocumented immigrants may find it difficult to go through the immigration court process if there's more people in queue. So most likely, they may decide to return back to their country if that's the case.

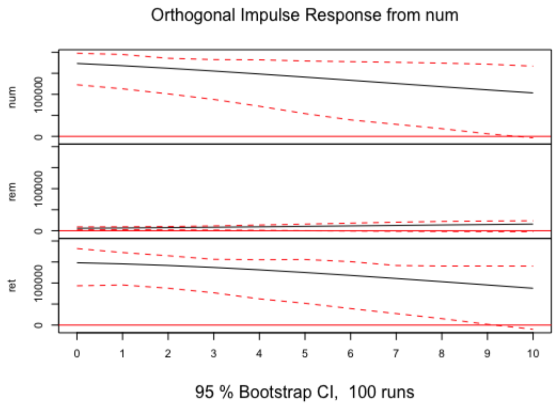


Figure 28: IRF of num

5.2 Portmanteau Test

The Portmanteau test for VAR models is similar to the Box-Ljung test for ARIMA models. The Portmanteau test determines if the VAR model's residuals are white noise. Here's the conditions of the test:

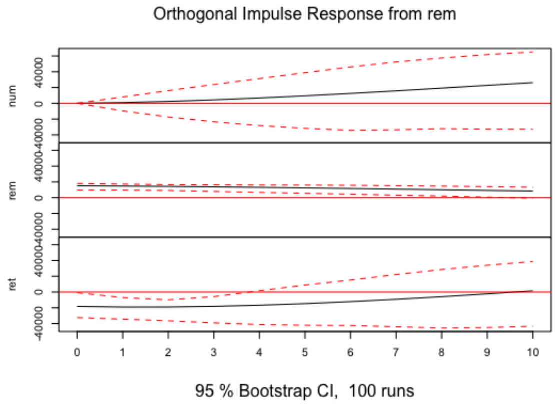


Figure 29: IRF of rem

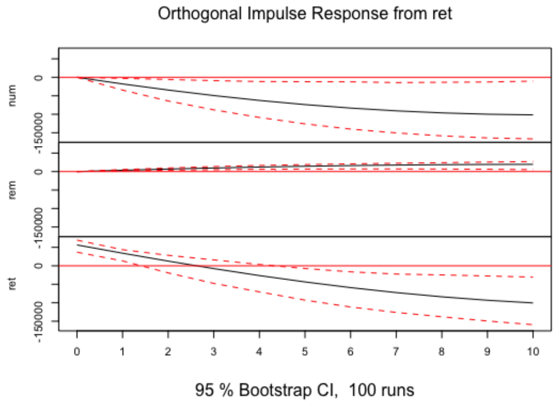


Figure 30: IRF of ret

- $\underline{H_0}$: Variable residuals are not serially correlated (white noise)
- $\underline{H_a}$: Variable residuals are serially correlated (not white noise)
- Condition: Reject H_0 if p-value is less than 0.05. This means if the p-value is greater than 0.05 the residuals are white noise and the variables are not serially correlated.

Since the p-value of the VAR(1) model is 0.4004, the residuals are deemed white noise and the variables are not serially correlated.