Time Series Analysis of Chicago Homicide Frequency

Victor Cardeno, Caroline Hills, Nicholas Milikich, & Laura Patterson

Abstract

Chicago is a large bustling city with a constant flow of professionals and tourists; it is also a city with a notorious crime rate. The City of Chicago publishes homicide offenses and their respective time of occurence and location on their website. The objective of this study is to analyze this data on Chicago homicides to determine if we can develop a time series model that would accurately forecast the most likely time of future homicides. Significant predictions of trends or seasonality in homicide rates might allow law enforcement to prepare to respond or enact preventative measures. Initial analysis and visualization of the data shows that homicide consistently peaks in the summer months when the days are longer, people are outside more often, and tourists visit the city for the various summer events. After running our analysis, we concluded that a seasonal ARIMA$(2,1,1)(2,1,2)_{12}$ model was the best model to predict the number of homicides each month in the Chicago city limits. Our conclusions could help the City of Chicago and its law enforcement departments better focus their workforce and, hopefully, reduce crime in the city for the benefit of its citizens and visitors.

Introduction

This report is an analysis of the published history of homicides in the city of Chicago. At the end of our analysis, we expect to form conclusions about the patterns of crime in the Chicago area, such as the months where crime is most common or show a significant increase or decrease in crime. The results gleaned from this report could potentially help to identify specific points in time that prove to be particularly dangerous in the city, informing the police department and aiding in their efforts to fight wrongdoing in the community. It could also serve to contextualize the data;

if our conclusions highlight a month or year where an upward or downward trend in homicides began, it could point researchers to look more closely at any events or legislation that occurred around that time that may have contributed to the change in pattern. We believe this is an important research topic due to the background of the city of Chicago. It is a wonderful city laden with attractions and rich in history, but its standing is weighed down by significant issues with crime and delinquency. Remedying this problem would go a great way towards improving the lives of those in the city and its surrounding area.

Data

The dataset includes an entry for each recorded homicide in the city of Chicago from January 2001 through September 2019. Each row describes one particular incident along with qualitative and quantitative details regarding the incident, such as the case number, general location, FBI code, and more. The data is sourced from the Chicago Police Department's Citizen Law Enforcement Analysis and Reporting system. A sample of the data is shown in Figure 1. In order to conduct a time series analysis on the data, we needed the frequencies of homicides committed in each month of each year. We manipulated the data by extracting the month and day of each incident from the date column and then creating a dataframe with each year as a row and each month as a column, allowing us to record the total number of incidents occurring in Chicago by month and year in the variable *hom.dummy*. We then fit the *hom.dummy* dataset to a time series, named *hom.ts*, to run our models and create our predictions. The data were split into a training set (January 2001 through September 2017) and test set (October 2017 through September 2019).

To begin our analysis, we produced some basic graphs to help us gain a better understanding of the data. We started with a basic plot of the time series data, graphing number of monthly homicides through time. This plot is shown in Figure 2. From this figure, we suspect that

there is a seasonal aspect to the data, as there are regularly spaced spikes and dips. There appeared to be neither a dominant trend nor a cyclicality. The variance appears constant, so a transformation is not necessary. Next, we created a seasonal plot and a subseries plot of the homicide data in order to further examine the potential seasonality. These plots are shown in Figures 3 and 4. Both of these plots confirmed the seasonality we suspected to be present, as they both showed that the number of homicides was consistently higher in the months between May and November. Because of the apparent seasonality, a monthly seasonal differencing was performed and did increase the stationarity of the data (figure 5), and an additional non-seasonal differencing further improved the stationarity (figure 6). Lastly, we created ACF and PACF plots, which are shown in Figures 7 and 8. The ACF plot also confirms our impressions of the data from the initial plot, showing seasonality without trend or cyclicality. The alternating pattern and spikes of decreasing magnitudes in the PACF plot suggest that a moving average term is present in the data that should be explored using an ARIMA model, and further confirms the suspected seasonality with a lack of significant spikes past the twelfth lag. Going forward, we know that we will need to account for seasonality when building our models.

Data Analysis

Linear Regression (Figure 9)

$$\hat{y} = 33.25 + .0004351*t - 7.236*Feb + 1.470*Mar + 6.234*Apr + 12.29*May + 19.41*Jun +$$
$$23.88*Jul + 19.88*Aug + 15.94*Sep + 10.95*Oct + 6.892*Nov + 3.954*Dec$$

The basic linear model illustrated the magnitude of potential linear relationships within the data. The model is set with trend and season as predictors. The most significant predictors are seasons 6-9, corresponding to the months June - September. Their coefficients are positive; the expected number of homicides will be, on average, higher in those months than in January (the

baseline). This was what we originally hypothesized given the longer days and large amounts of tourists enjoying Chicago events in those months.

The ACF plot of the linear model shows significant trend and the Breusch-Godfrey test has a p-value of virtually 0. We conclude that there are correlations among the lags. The Ljung-Box test suggests the presence of significant remaining information and possible need for a more complex model with a p-value of essentially 0. The test set RMSE of the linear model is 8.00 and the AICc is 1016.76.

Linear Regression with ARIMA Errors (Figure 10)

$$y_t = 0.4823temp_t + \eta_t$$

$$(1 - 0.0038B^{12} - 0.1459B^{24})(1 - B)\eta_t = (1 - 0.7396B)\varepsilon_t$$

The linear regression model with ARIMA errors was fit using average monthly temperature as a predictor. The resulting model ACF graph showed that the residuals are white noise, and the Ljung-Box test p-value is 0.98, hence the model is sufficient. The test set RMSE and AICc of the model are 8.71 and 1445.78, respectively.

ARIMA (figure 11)

$$(1 + 0.1596B + 0.0945B^2)(1 + 0.3743B^{12} + 0.0213B^{24})(1 - B)(1 - B^{12})y_t$$

$$= (1 - 0.6297B)(1 - 0.597B^{12} - 0.2255B^{24})\varepsilon_t$$

The homicide data required seasonal differencing and first order differencing to obtain stationality. The R-fitted ARIMA model was an ARIMA(2,1,1)$(2,1,2)_{12}$ model. The ACF plot showed only one significant lag at l = 35, and the Ljung-Box test returned a p-value of 0.89. Therefore, we fail to reject that the residuals are not autocorrelated; there are no patterns evident and the model is sufficient. The test set RMSE is 14.79 and the AICc is 348.31.

ETS (figures 12 & 13)

$$\hat{y}_{t+h|t} = l_t + s_{t+h-12(k+1)}$$

$$l_t = 0.2576(y_t - s_{t-12}) + (1 - 0.2576)l_{t-1}$$

$$s_t = 0.0001(y_t - l_{t-1}) + (1 - 0.0001)s_{t-12}$$

We fit an ETS(M,N,A) model of the data: multiplicative error, no trend, and additive seasonality with parameters α = 0.2576 and γ = 0.0001. An analysis of the residuals suggests that no patterns are evident (Ljung-Box test p-value = 0.20). The test set RMSE is 13.59 and the AICc is 1949.17.

The data was also fitted with the Holt-Winters additive function to see if any improvements could be made on the model and the results were similar to the ETS model. The residuals are statistically white noise (Ljung-Box test p-value = 0.33), the test set RMSE is 14.32 and the AICc is 1963.63.

Neural Network (figure 14)

The fitted neural network is NNAR(12,1,6)$_{12}$. The ACF plot shows one significant lag that just barely breaks the interval and the p-value from the Ljung-Box Test is 0.93; therefore, the residuals could justly be considered white noise and the model sufficient. The RMSE for the test data set is 13.74 and AICc was unable to be calculated.

Discussion

We gained valuable insights from the linear regression. We originally attempted to use average monthly temperature as a predictor in the tslm() function. The resulting linear model did not line up with the data, so we decided to leave out the predictor for the tslm() model. The model did note that the months of May through October are the most significant predictors for homicide numbers. However, we wanted to include the temperature data to test our hypothesis that warmer

weather did impact the number of homicides; we included this data as a predictor in a regression model with ARIMA errors.

Using external temperature data as a regressor in a linear model with ARIMA errors created a model with white noise residuals. The temperature data was taken from the National Center for Environmental Information; one value was missing and was filled by averaging the surrounding data points. When graphed, the model does an excellent job staying true to the peaks of the test set data, but fails to account for the valleys in the test set. The model does not account for trend as well as the regular linear model, but the trend was deemed insignificant at an alpha level of 0.05. It is likely that, in order to get a better idea of the temperature trend in the data, you could break down the average temperature into smaller time intervals, such as weekly or even daily recordings and look at the number of homicides at that respective granularity too.

The ARIMA model required seasonal and regular differencing to obtain stationarity. An ADF test of the complete differenced data returned a p-value of 0.043. When fit by the auto.arima() function, the data was transformed with a Box-Cox lambda of 0.26. The resulting RMSE of the ARIMA model is surprising as it indicates that the linear model, which was deemed insufficient in the Ljung-Box test, performed much better on the test data than the more complex ARIMA model.

The ETS model had no significant predictive power or insights that would cause us to choose it as our final model over the others. We decided to fit a Holt-Winter's additive model, because the data appeared to have roughly constant variations, to see if we could gain any predicting power, but the resulting model had slightly higher AICc and RMSE values that indicated it was not better than the ETS model.

Neural network models tend to be robust to extreme data, which might make them a good candidate to capture the spike in the data around 2016 and was our motivation to test this forecasting method. The neural network model had a RMSE relatively similar to the other models and the residuals are white noise, but there is no other evidence to show that this model would be a better predictor over the other models. It is especially difficult to choose this model without having an AICc to compare.

	TSLM	**TSLM ARIMA Errors**	**ETS**	**Neural Network**	**HW**	**ARIMA**
AICc	1016.76	1445.78	1949.17	N/A	1963.63	348.31
RMSE	8.00	13.99	13.59	13.74	14.32	14.79
MAPE	14.20	31.20	30.62	30.31	32.12	31.92

Table 1: Performance metrics of each model on the test set

Table 1 shows the AICc, RMSE, and MAPE of the tested models. RMSE and MAPE performance on the test set suggests that the model with by far the best predictive power was the linear model. However, the linear model is the only model that has non-white noise residuals, so it was not considered as a possible final model.

It is surprising that the more complex models performed so much worse on the test set, especially considering the fact that the linear model, which gave by far the most visually appealing forecasts compared to the test data, was insufficient based on residual analysis. The data shows a spike in 2016, followed by a decrease in 2017 and a return to normal levels in 2018. It appears that all the complex models captured this downward trend in the data that did not actually continue, while the linear model did not. Therefore, the observed model performances may be due to quirks

in this particular data set. In further study, this might be corrected by comparing model performance using test sets of variable lengths. For this reason we gave more weight to the AICc than the test set performance in choosing a final model.

The data does have a trend, but visually it does not seem to be more than slight. We think the seasonality in the data is the more important quality to pay attention to. The data is strongly seasonal and incorporating this aspect into the final model will lead to better predicting power. Therefore, we decided the best model to choose as our final model was the ARIMA model because it captured the seasonality best, had white noise residuals, and had a significantly lower AICc than the other models.

Other variables to account for in future analysis include changes in policy, national societal phenomena, or relevant events in the City of Chicago. It would also be beneficial to include external data on total hours of daylight, unemployment rate, and other statistics that could impact motivations for homicide. For instance, would highly publicized instances of police brutality cause fear and decrease homicides or incite reactionary actions that boost the homicide numbers? It is also important to note, especially in forecasting future crime, that this dataset only includes crime that was reported and investigated by the police. Especially in a large city like Chicago where underground gang violence is prevalent, it is quite likely that this dataset does not provide a wholly accurate assessment of crime in the area.

Our conclusions and insights from this analysis hold a substantial amount of value when investigating the City of Chicago and its crime rates. It could help the police force in the city by informing them about peaks and troughs in crime throughout the year, allowing them to prepare their personnel accordingly and optimize resource allocation. It could also serve the citizens of Chicago through its societal implications. If the municipal government is aware of trends in crime

it can proactively provide safety information to citizens and create a safer environment, which generates a positive externality that benefits people living in and visiting the area. In addition, decreasing crime could help to reverse the city's reputation as one of the most dangerous cities in the United States. Although Chicago already experiences great tourism and events, a shift in public opinion opens many new and unimaginable opportunities for the Windy City.

References

1. Chicago Data Portal. Homicides https://data.cityofchicago.org/Public-Safety/Homicides/k9xv-yxzs
2. NOAA National Centers for Environmental Information. Climate at a Glance: City Time Series https://www.ncdc.noaa.gov/cag/

Appendix

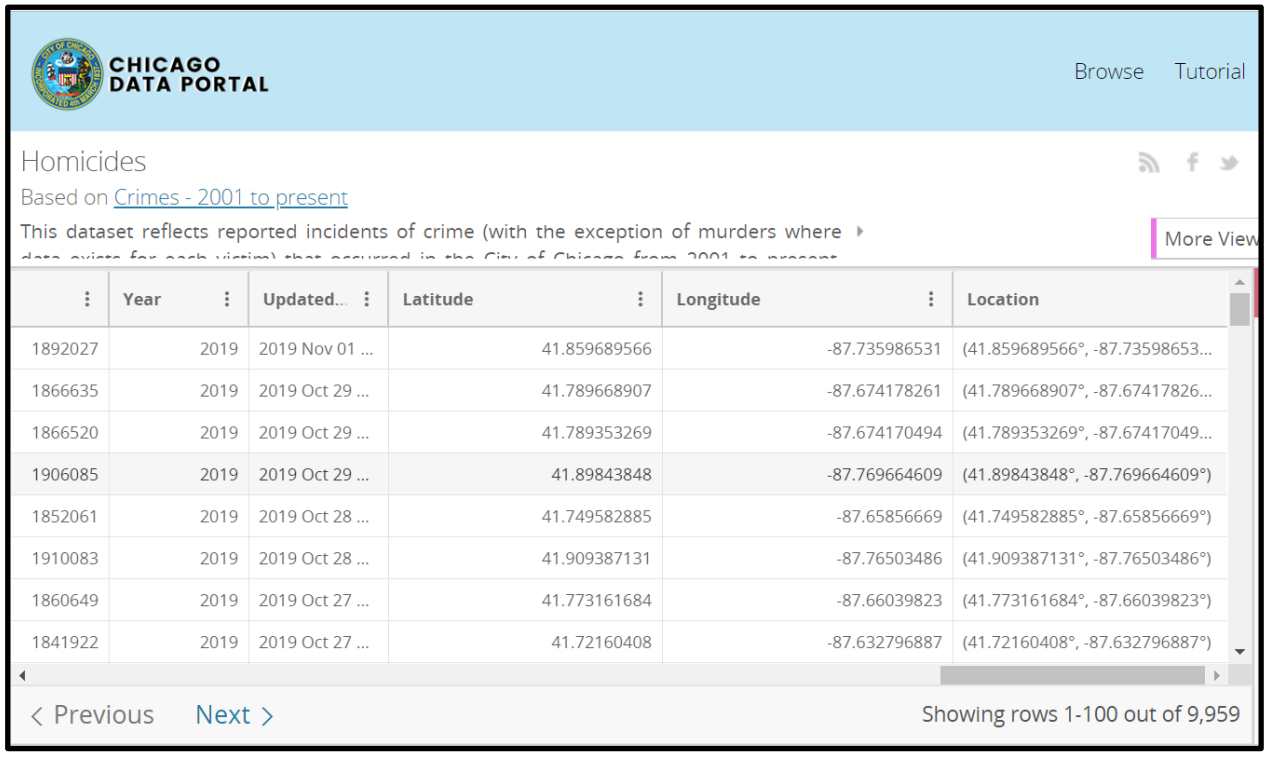


Figure 1: Website for the Chicago Police Department's Citizen Law Enforcement Analysis and Reporting system

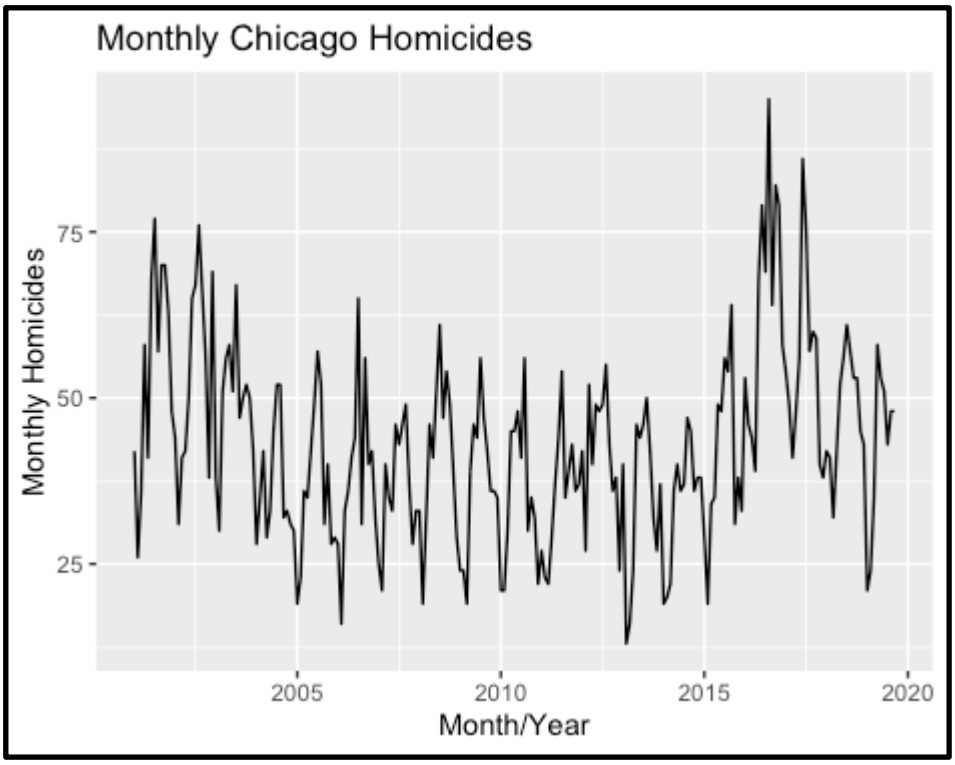


Figure 2: Simple time series plot after extracting monthly count data

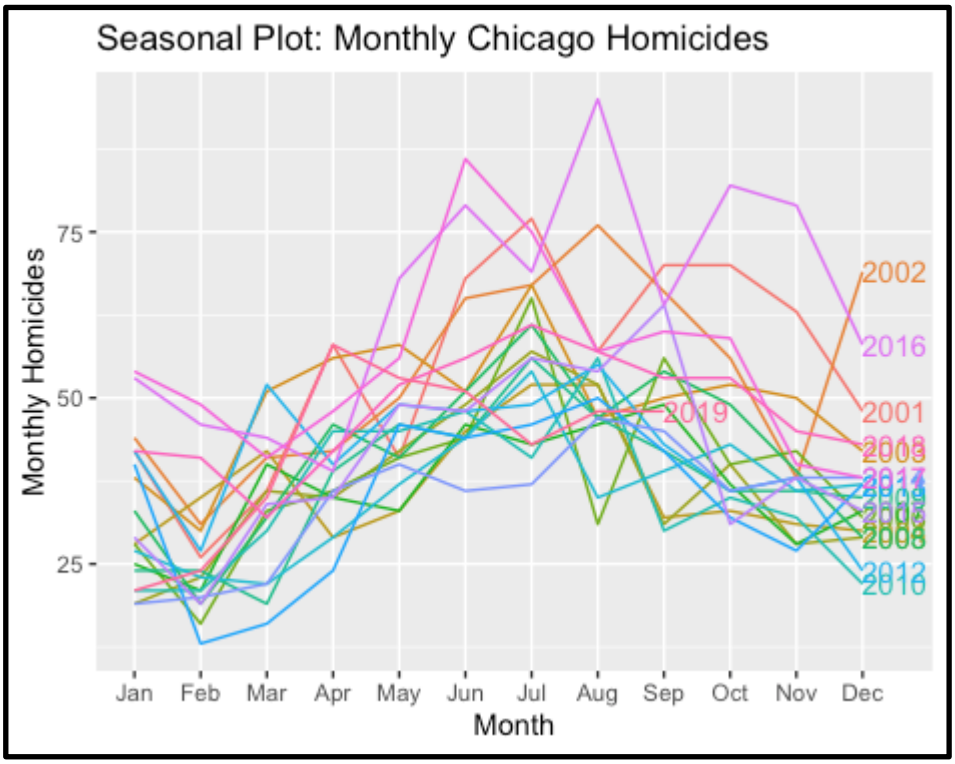


Figure 3: Seasonal plot of the data

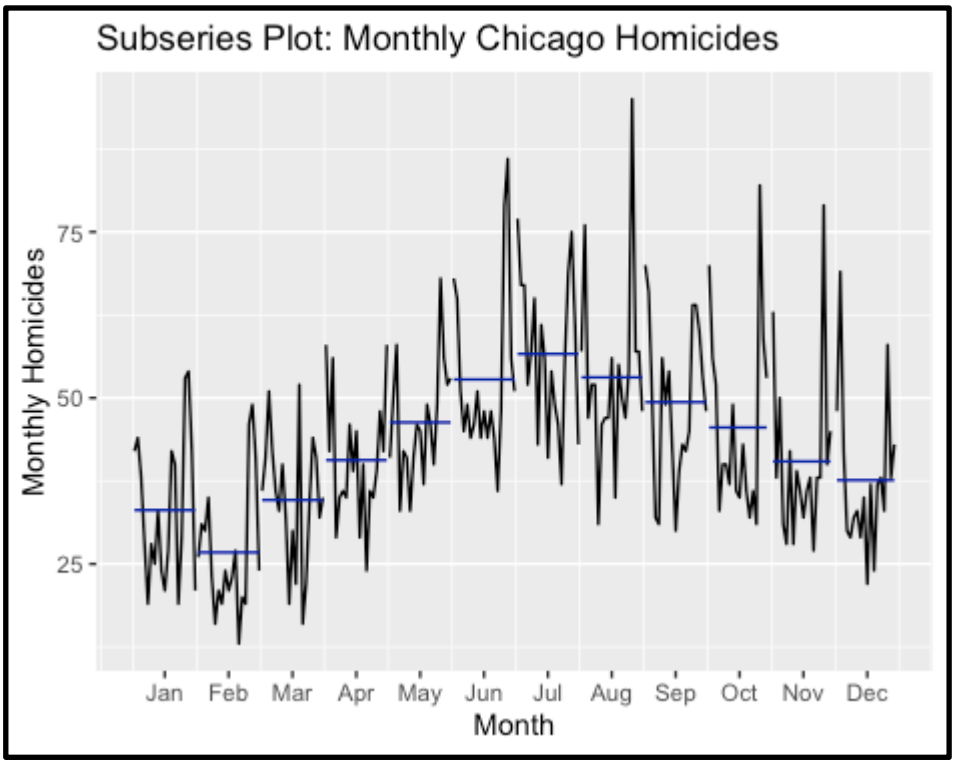


Figure 4: Subseries plot of the data

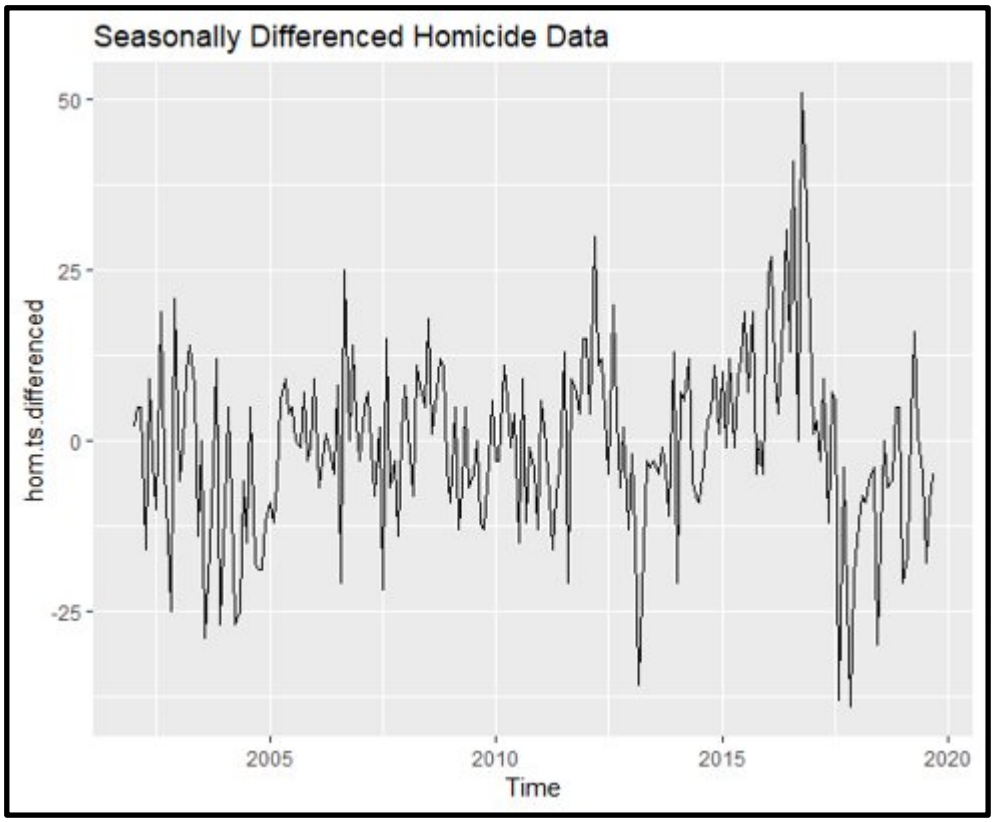


Figure 5: Time series plot of the seasonally differenced time series (lag 12)

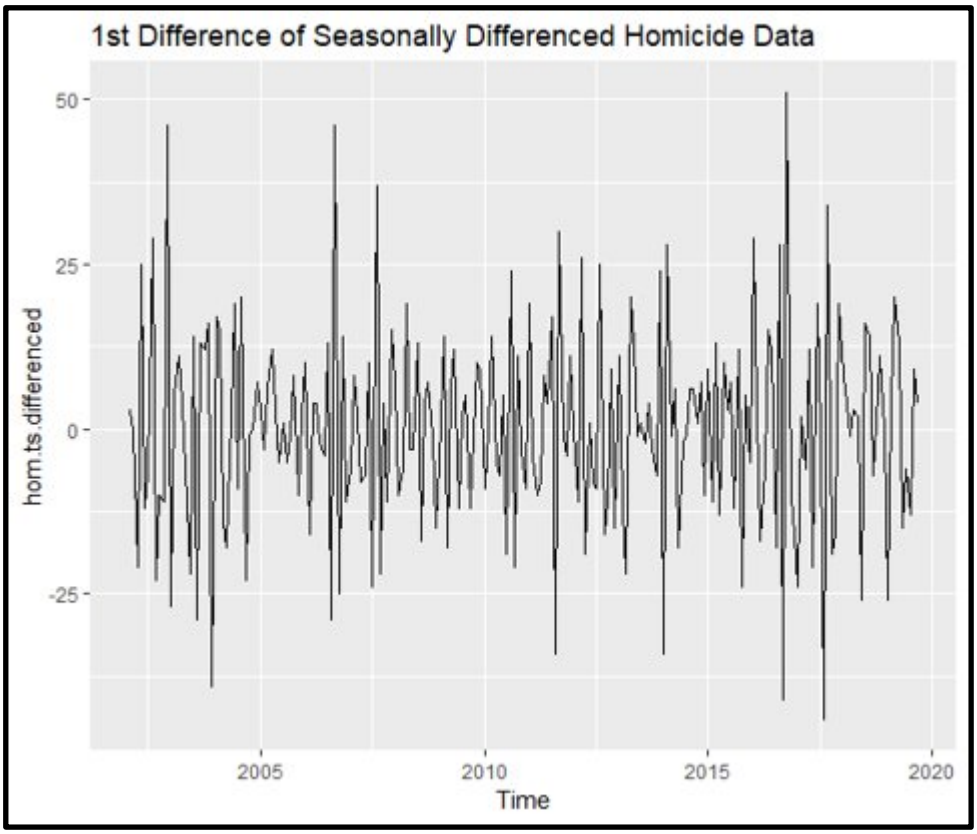


Figure 6: Time series plot of the data after applying seasonal differencing (lag 12) and differencing (lag 1)

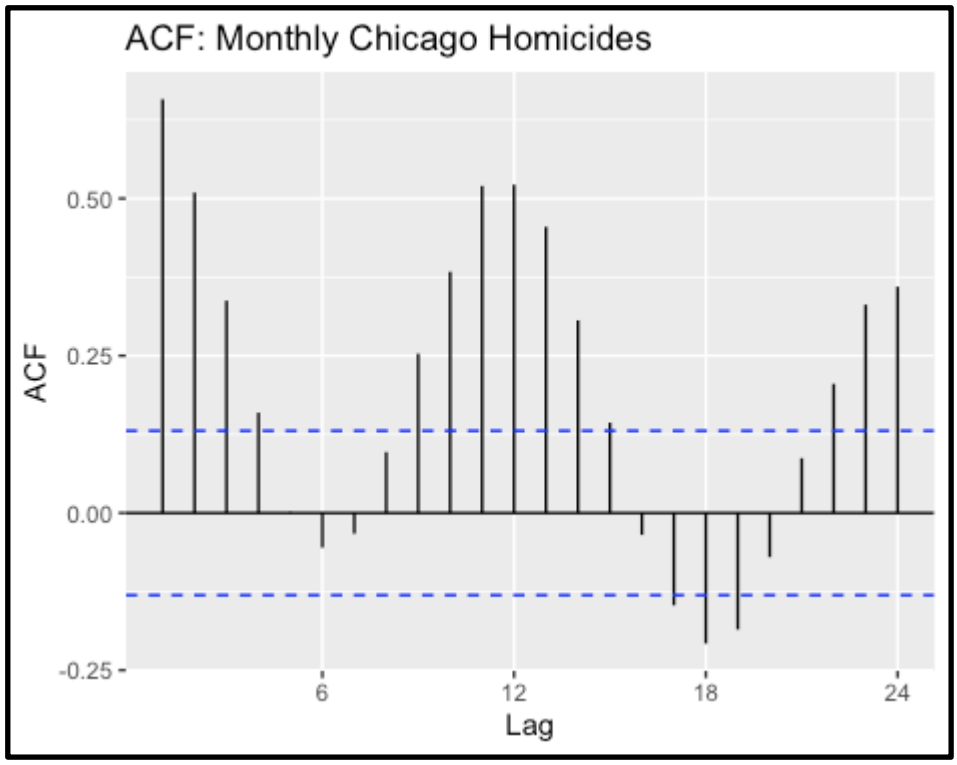


Figure 7: ACF of the raw homicide data

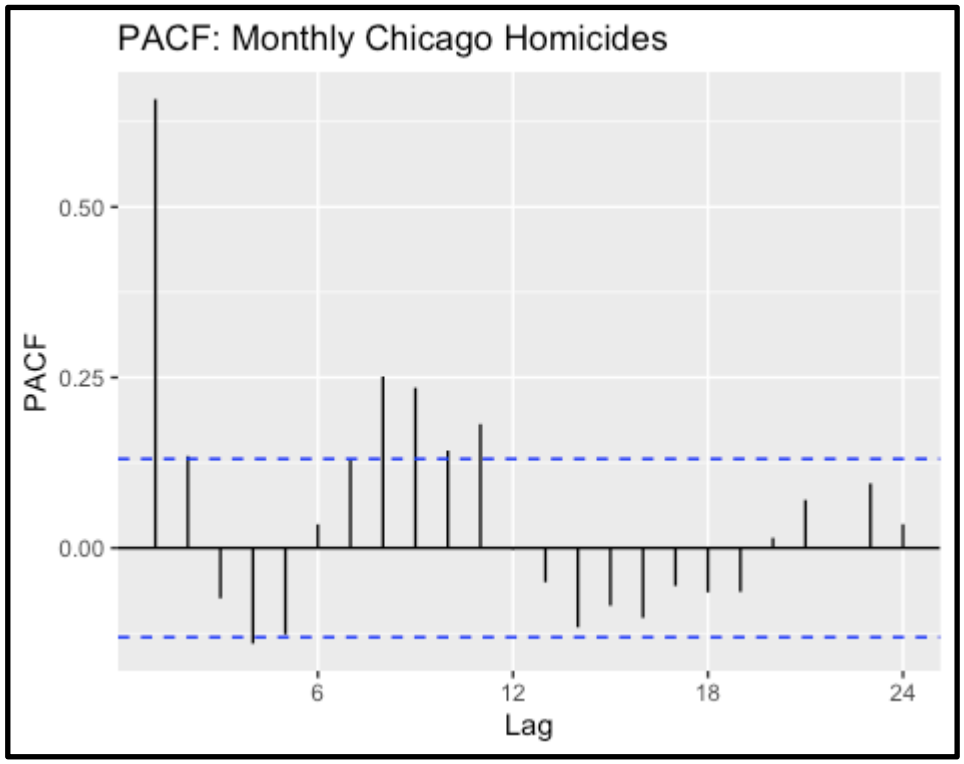
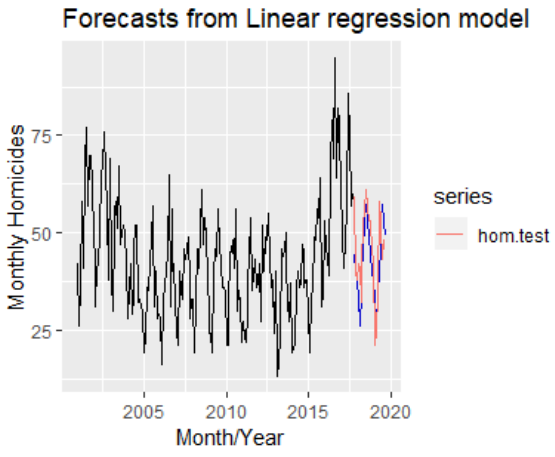


Figure 8: PACF of the raw homicide data

```
Call:
tslm(formula = hom.train ~ trend + season)

Residuals:
    Min      1Q  Median      3Q     Max
-22.161  -7.256  -2.221   5.465  41.787

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 33.2519093  3.2447551  10.248  < 2e-16 ***
trend        0.0004351  0.0146355   0.030 0.976313
season2     -7.2357293  4.1262965  -1.754 0.081135 .
season3      1.4697180  4.1263743   0.356 0.722108
season4      6.2339887  4.1265041   1.511 0.132539
season5     12.2923771  4.1266858   2.979 0.003276 **
season6     19.4095890  4.1269194   4.703 4.95e-06 ***
season7     23.8797421  4.1272048   5.786 2.97e-08 ***
season8     19.8793070  4.1275421   4.816 3.00e-06 ***
season9     15.9376954  4.1279313   3.861 0.000155 ***
season10    10.9545769  4.1904776   2.614 0.009670 **
season11     6.8916418  4.1906565   1.645 0.101739
season12     3.9537067  4.1908865   0.943 0.346684
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 12.03 on 188 degrees of freedom
Multiple R-squared:  0.3712,    Adjusted R-squared:  0.3311
F-statistic:  9.25 on 12 and 188 DF,  p-value: 6.31e-14
```

```
                      ME      RMSE      MAE       MPE      MAPE      MASE      ACF1 Theil's U
Training set -1.945652e-16 11.634509 8.886282 -7.528211 22.49666 0.9005401 0.5655587        NA
Test set      2.807971e+00  8.001407 6.141633  3.613329 14.20224 0.6223960 0.2140498 0.7528447
```



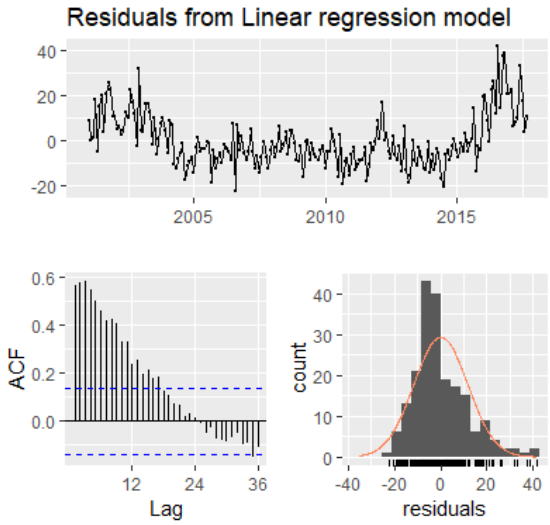


Figure 9: Output from Linear Regression Model

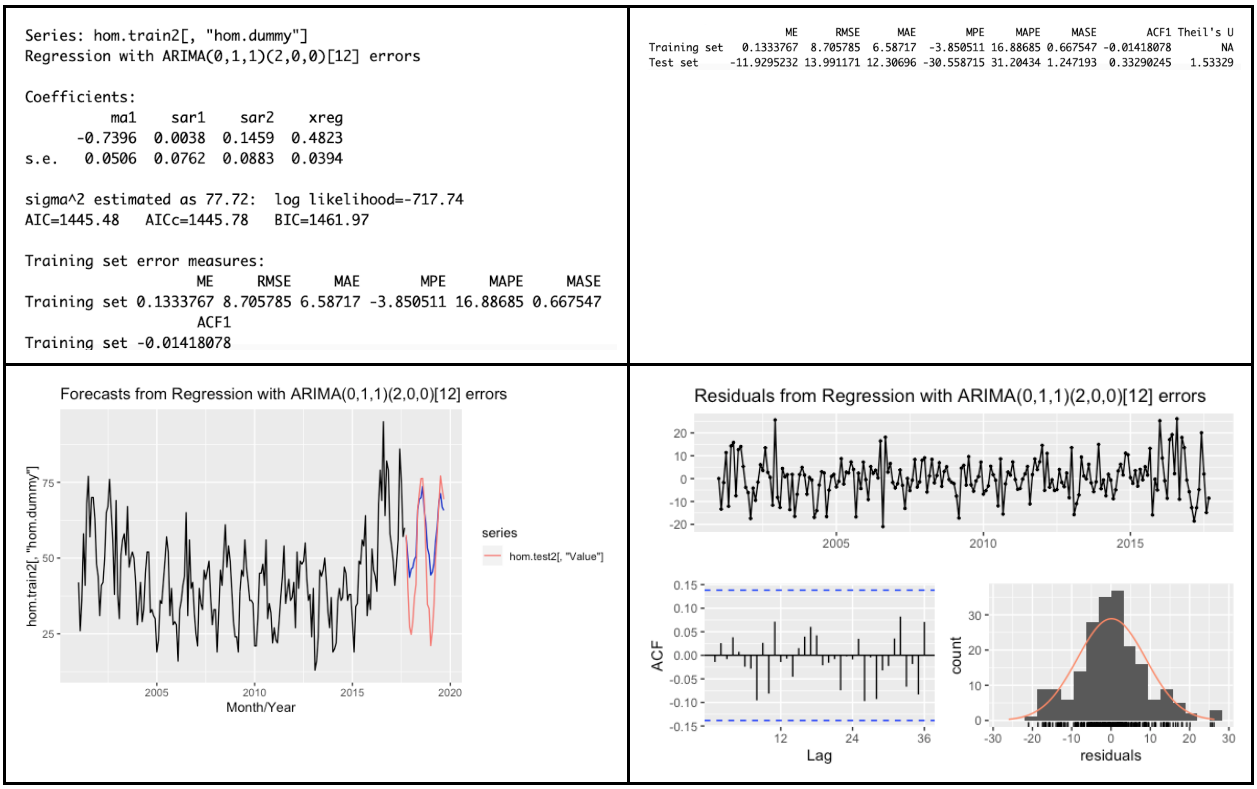


Figure 10: Output from Regression Model with ARIMA errors

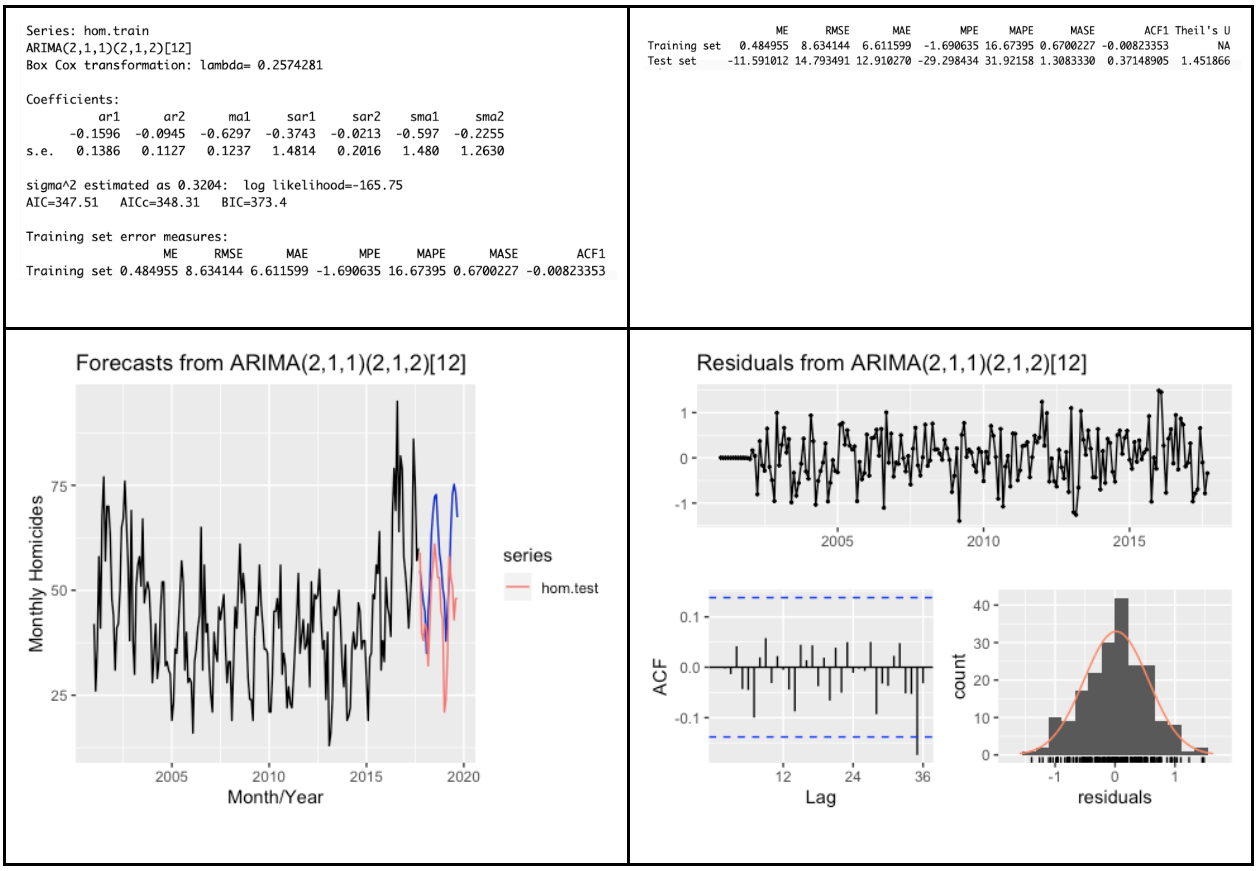
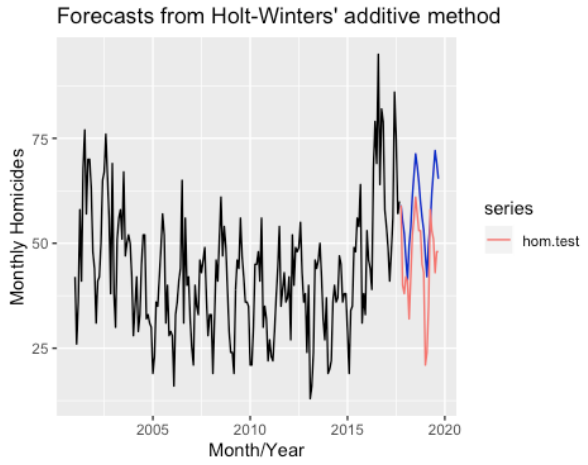


Figure 11: Output from ARIMA Model

```
Forecast method: Holt-Winters' additive method

Model Information:
Holt-Winters' additive method

Call:
 hw(y = hom.train, seasonal = "additive")

  Smoothing parameters:
    alpha = 0.2657
    beta  = 1e-04
    gamma = 1e-04

  Initial states:
    l = 53.2883
    b = 0.0683
    s = -4.8726 -1.8279 1.965 6.7932 10.5767 13.6349
           8.7204 3.6922 -3.6588 -8.0058 -16.0884 -10.9289

  sigma:  8.8602

     AIC     AICc      BIC
1960.282 1963.627 2016.439

Error measures:
                     ME     RMSE      MAE       MPE     MAPE      MASE         ACF1
Training set -0.1629808 8.500243 6.604516 -4.001448 16.81223 0.6693049 -0.06375353
```

```
                     ME     RMSE      MAE       MPE     MAPE      MASE         ACF1 Theil's U
Training set -0.1629808 8.500243 6.604516 -4.001448 16.81223 0.6693049 -0.06375353        NA
Test set     -12.2078529 14.315660 12.491444 -31.626094 32.11504 1.2658889  0.24440935  1.537043
```



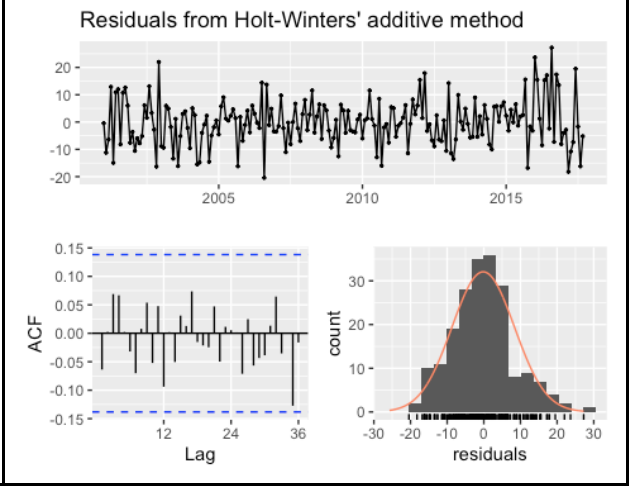


Figure 12: Output from Holt-Winters Model with Additive Seasonality

```
ETS(M,N,A)

Call:
 ets(y = hom.train)

  Smoothing parameters:
    alpha = 0.2576
    gamma = 1e-04

  Initial states:
    l = 54.9973
    s = -5.4537 -2.9402 -0.5458 6.3106 10.8194 13.1709
           9.2953 2.9496 -1.7162 -6.8487 -15.6563 -9.3849

  sigma:  0.21

     AIC     AICc      BIC
1946.577 1949.172 1996.127

Training set error measures:
                     ME     RMSE      MAE       MPE     MAPE      MASE         ACF1
Training set 0.03760055 8.555476 6.64467 -3.917581 16.93583 0.673374 -0.04196138
```

```
                     ME     RMSE      MAE       MPE     MAPE      MASE         ACF1 Theil's U
Training set  0.03760055 8.555476 6.64467  -3.917581 16.93583 0.673374 -0.04196138        NA
Test set     -11.31889780 13.589325 11.76655 -29.858832 30.62432 1.192428  0.20854910  1.48215
```

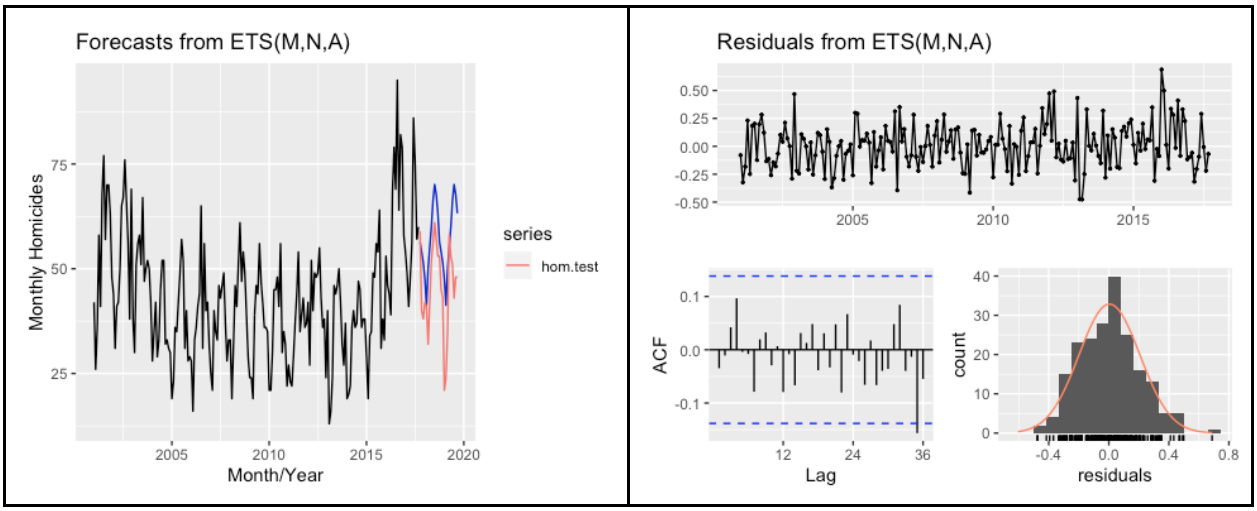


Figure 13: Output from ETS Method

```
Series: hom.train
Model:  NNAR(12,1,6)[12]
Call:   nnetar(y = hom.train)

Average of 20 networks, each of which is
a 12-6-1 network with 85 weights
options were - linear output units

sigma^2 estimated as 9.15
```

	ME	RMSE	MAE	MPE	MAPE	MASE	ACF1	Theil's U
Training set	-0.004229447	3.024864	2.270017	-1.749767	6.368765	0.2300446	0.006026261	NA
Test set	-10.078100165	13.743721	11.979979	-26.409987	30.308917	1.2140568	0.354402614	1.608816

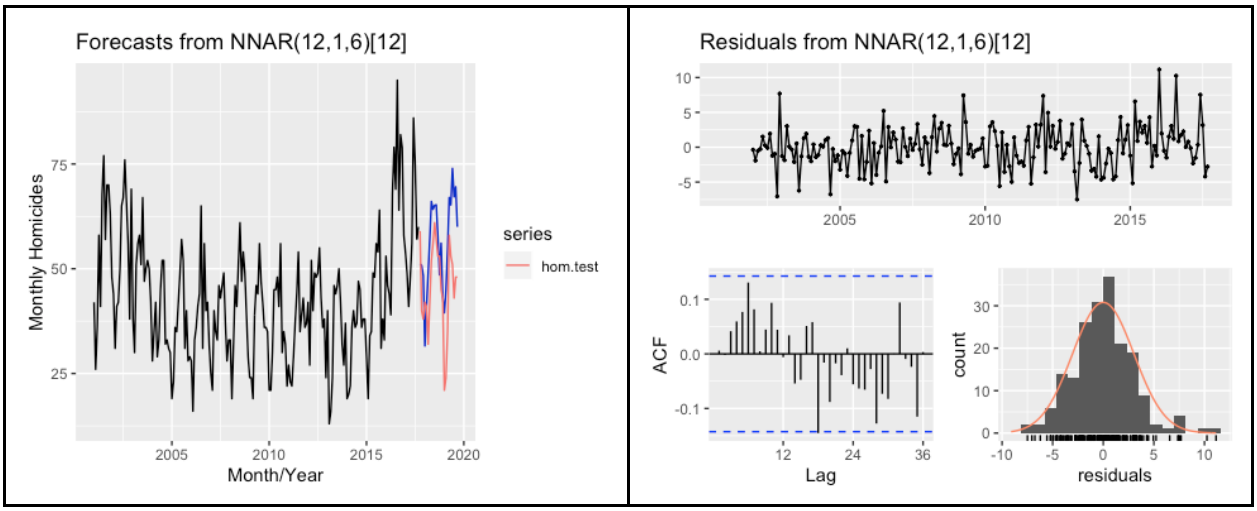


Figure 14: Output from Neural Network Method

R Code

```
############################
# Data import and processing
############################

setwd("~/Google\ Drive\ (nmilikic@nd.edu)/Fall\ 2019/ACMS\
40842\ Time\ Series\ Analysis/Time\ Series\ Project/")
hom = read.csv(file = "Homicides (1).csv", header = TRUE, sep =
",")
hom.dummy = numeric(0)
currMonth = 1
currYear = 2001
currVal = 0
for (i in seq(length(hom$Date),1,-1))
{
  month = as.numeric(substr(toString(hom[i,"Date"]),0,2))
  year = as.numeric(substr(toString(hom[i,"Date"]),7,10))
  if (month == currMonth & year == currYear)
    currVal = currVal + 1
  else
  {
    hom.dummy = c(hom.dummy, currVal)
    currVal = 0
    if (currMonth == 12)
    {
      currMonth = 1
      currYear = currYear + 1
    }
    else
      currMonth = currMonth + 1
  }
}

hom.ts = ts(hom.dummy, frequency = 12, start = c(2001,1))


###############
# Visualization
###############
```

```
library(fpp2)
autoplot(hom.ts) + xlab("Month/Year") + ylab("Monthly
Homicides") + ggtitle("Monthly Chicago Homicides")
ggseasonplot(hom.ts, year.labels = TRUE) + ylab("Monthly
Homicides") + ggtitle("Seasonal Plot: Monthly Chicago
Homicides")
ggsubseriesplot(hom.ts) + ylab("Monthly Homicides") +
ggtitle("Subseries Plot: Monthly Chicago Homicides")
ggAcf(hom.ts) + ggtitle("ACF: Monthly Chicago Homicides")
pacf(hom.ts)

hom.train = window(hom.ts, end = c(2017,09))
hom.test = window(hom.ts, start = c(2017,10))

##############################
# Time Series Regression Model
##############################

hom.tslm <- tslm(hom.train~trend+season)
summary(hom.tslm)
# The season coefficients show a peak at season 7 (July),
suggesting that season impacts the number of homicides
#   The seasons where temperature is normally higher (May
through September/October) are positive
#   meaning that the number of homicides is expected to increase
during those months.
#Therefore, our initial attempt to analyze the linear model with
external Temperature data was unnecessary

autoplot(forecast(hom.tslm, h = 24), PI = FALSE) +
autolayer(hom.test) + xlab("Month/Year") + ylab("Monthly
Homicides")
checkresiduals(hom.tslm)
# Acf plot shows strong trend

#Breusch Godfrey Test
# p-value: 2.062e-12 -> there is some correlation of the lags

accuracy(forecast(hom.tslm, h = 24), hom.test)
#RMSE (test): 8.001407
library(fpp)
```

```
CV(hom.tslm) #1016.756

#############
# TSLM ARIMA ERRORS
#############
temp = read.csv(file = "Temperature.txt", header = TRUE, skip =
4, sep = ",")
temp$Value[143] <- 0.5*(temp$Value[142]+temp$Value[144])
temp$Anomaly[143] <- 0.5*(temp$Anomaly[142]+temp$Anomaly[144])
temp.ts = ts(temp$Value, frequency = 12, start = c(2001,1))
# temp.ts[143] is missing data that has been replaced as the
average between temp.ts[142] and temp.ts[144]
temp.train = window(temp.ts, end = c(2017,09))
temp.test = window(temp.ts, start = c(2017,10))
temp = temp[-226,]

hom.dumm2 <- cbind(hom.dummy, temp)
hom.ts2 = ts(hom.dumm2, frequency = 12, start = c(2001,1))
hom.train2 <- window(hom.ts2, end = c(2017,09))
hom.test2 = window(hom.ts2, start = c(2017,10))

hom.arimaerrors <- auto.arima(hom.train2[,"hom.dummy"], xreg =
hom.train2[,"Value"])
summary(hom.arimaerrors)
#ARIMA(0,1,1)(2,0,0)[12]
#AICc = 1445.78
#RMSE = 8.705785
autoplot(forecast(hom.arimaerrors, xreg = hom.test2[,"Value"], h
=24),PI = FALSE) + autolayer(hom.test2[,"Value"]) +
xlab("Month/Year")
ggtsdisplay(residuals(hom.arimaerrors, type="response"), main =
"Regression Errors")
checkresiduals(hom.arimaerrors)
CV(hom.arimaerrors)

#############
# ARIMA Model
#############

hom.arima = auto.arima(hom.train, lambda =
BoxCox.lambda(hom.ts))
```

```
summary(hom.arima)
#ARIMA(2,1,1)(2,1,2)[12]
# This model has a first order differencing with a seasonal lag
of m = 12
autoplot(forecast(hom.arima, h = 24), PI = FALSE) +
autolayer(hom.test) + xlab("Month/Year") + ylab("Monthly
Homicides")
checkresiduals(hom.arima)
# Significant lag at ~35
#LJung-Box test
# p-value = 0.8883 -> fail to reject that the time series is not
autocorrelated

accuracy(forecast(hom.arima, h = 24), hom.test)
#RMSE (test): 14.793524

###########
# Differencing Graphs
###########

hom.ts.differenced <- diff(hom.ts, lag = 12)
autoplot(hom.ts.differenced) + ggtitle("Seasonally Differenced
Homicide Data")
hom.ts.differenced <- diff(hom.ts.differenced)
autoplot(hom.ts.differenced) + ggtitle("1st Difference of
Seasonally Differenced Homicide Data")
autoplot(diff(hom.ts)) + ggtitle("1st Differenced Homicide
Data")
kpss.test(hom.ts.differenced) #p-val = 0.1

adf.test(hom.ts.differenced) #p-val = 0.04299
#ADF test states stationarity

###########
# ETS Model
###########

#The first model is Holt Winters Additive Seasonality
hom.fit1 = hw(hom.train, seasonal="additive")
summary(hom.fit1)
```

```
autoplot(forecast(hom.fit1, h = 24), PI = FALSE) +
autolayer(hom.test) + xlab("Month/Year") + ylab("Monthly
Homicides")
accuracy(hom.fit1)

#hom.fit2 = hw(hom.train, seasonal="multiplicative")
#summary(hom.fit2)
#autoplot(forecast(hom.fit2, h = 24), PI = FALSE) +
autolayer(hom.test) + xlab("Month/Year") + ylab("Monthly
Homicides")
#accuracy(hom.fit2)
#   The multiplicative seasonality was not needed. So we only
fit the additive method (which was the same as the ets())

#ETS Model
hom.ets <- ets(hom.train)
summary(hom.ets)
#ETS(M,N,A)
autoplot(forecast(hom.ets, h=24), PI = FALSE) +
autolayer(hom.test) + xlab("Month/Year") + ylab("Monthly
Homicides")
checkresiduals(hom.ets)
accuracy(forecast(hom.ets, h = 24), hom.test)


#######
# Neural Network Method
#######

library(forecast)
hom.neuron <- nnetar(hom.train, P = 12)
#NNAR(12,1,6)[12]
hom.neuron %>% forecast(h = 24, PI = FALSE) %>% autoplot() +
autolayer(hom.test) + xlab("Month/Year") + ylab("Monthly
Homicides")
checkresiduals(hom.neuron)
accuracy(forecast(hom.neuron,h=24), hom.test)
```