Forecasting-ARIMA

Chandan Gowda

9/24/2020

1. Introduction:

This assignment will look at a data "wmurders" from the fpp2 package which provides information on the number of women murdered each year per 100,000 standard population in the U.S

This assignment is interested in forecasting time series of "wmerders" data using ARIMA models which aims to describe the autocorrelations in the data.

Let us install the packages required and set a seed for reproducibility.

```r
if(!require("pacman"))
  install.packages("pacman")
```

```
## Loading required package: pacman
```

```
## Warning: package 'pacman' was built under R version 3.6.3
```

```r
pacman::p_load(fpp2, fpp3, ggplot2, rmarkdown, tidyverse, patchwork, purrr)
search()
```

```
##  [1] ".GlobalEnv"          "package:patchwork"   "package:forcats"
##  [4] "package:stringr"     "package:purrr"       "package:readr"
##  [7] "package:tidyverse"   "package:rmarkdown"   "package:fable"
## [10] "package:feasts"      "package:fabletools"  "package:tsibbledata"
## [13] "package:tsibble"     "package:lubridate"   "package:tidyr"
## [16] "package:dplyr"       "package:tibble"      "package:fpp3"
## [19] "package:expsmooth"   "package:fma"         "package:forecast"
## [22] "package:ggplot2"     "package:fpp2"        "package:pacman"
## [25] "package:stats"       "package:graphics"    "package:grDevices"
## [28] "package:utils"       "package:datasets"    "package:methods"
## [31] "Autoloads"           "package:base"
```

```r
theme_set(theme_classic())
options(digits = 3)
set.seed(42)
```

2. Studying the appropriate graphs of the series:

Let us explore the data for any trend or seasonality. The format of the data is Annual time series of class "ts" from 1950 to 2004

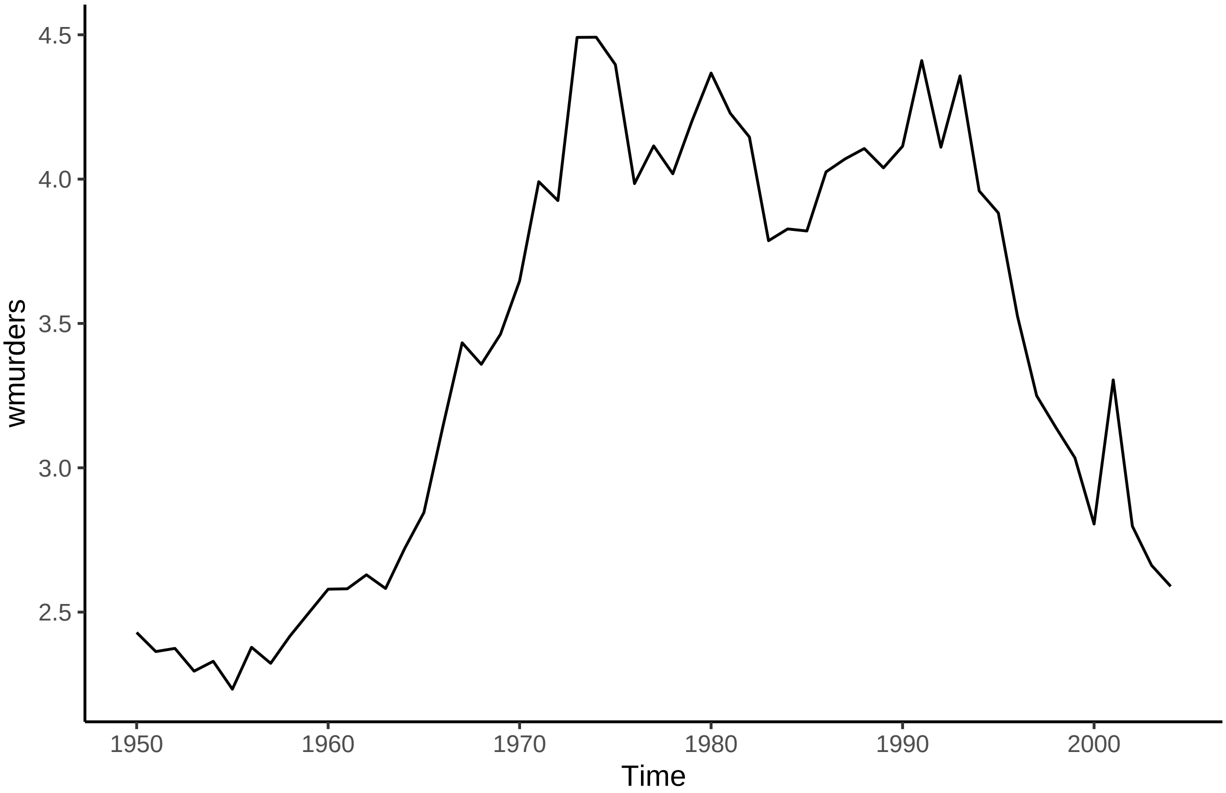
```r
str(wmurders)
```

```
##  Time-Series [1:55] from 1950 to 2004: 2.43 2.36 2.37 2.3 2.33 ...
```

```r
class(wmurders)
```

```
## [1] "ts"
```

```r
autoplot(wmurders)
```



From the above plot, it is clear that the data:

1. Do not need seasonal differencing because there is no seasonality present in the series
2. No Box-Cox transformation, as there is no significant variance in the time series through the different years

But the time series has a trend and the series looks like a Non-Stationary one as it is not horizontal and no constant variance present.

It looks like differencing is needed to make the data stationary which helps in stablizing the mean.

Lets us look at the KPSS test and ACF, PACF graphs to determine if differencing is required or not.

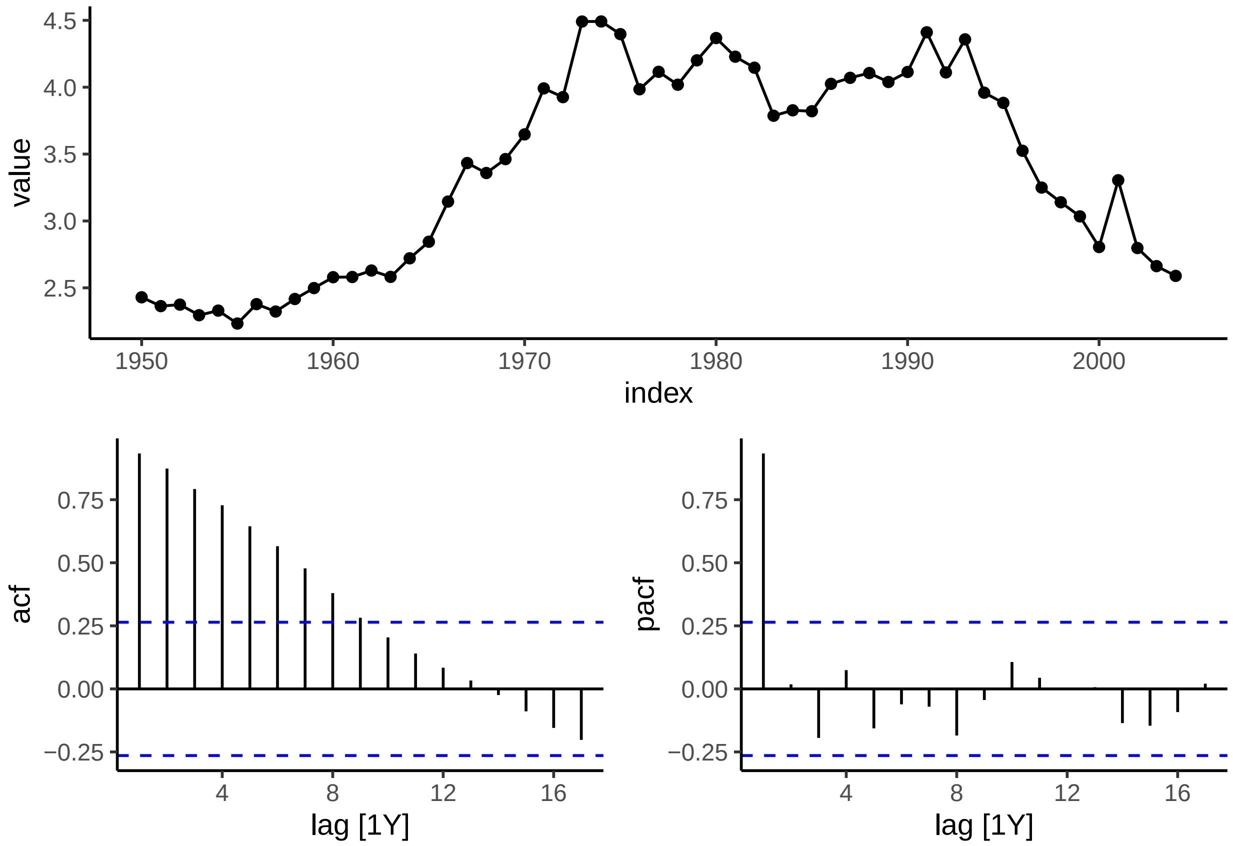
1. KPPS Test: Null Hypothesis in KPSS Test: The data is stationary From the test results, small p value of 0.0196 suggest that differencing is actually needed

```
wmurders %>% as_tsibble() %>% features(value, unitroot_kpss)
```

```
## # A tibble: 1 x 2
##   kpss_stat kpss_pvalue
##       <dbl>       <dbl>
## 1     0.633      0.0196
```

2. ACF and PACF plots: From the below plots, the time series is not seasonal but has a trend. Differencing is needed to make it stationary.

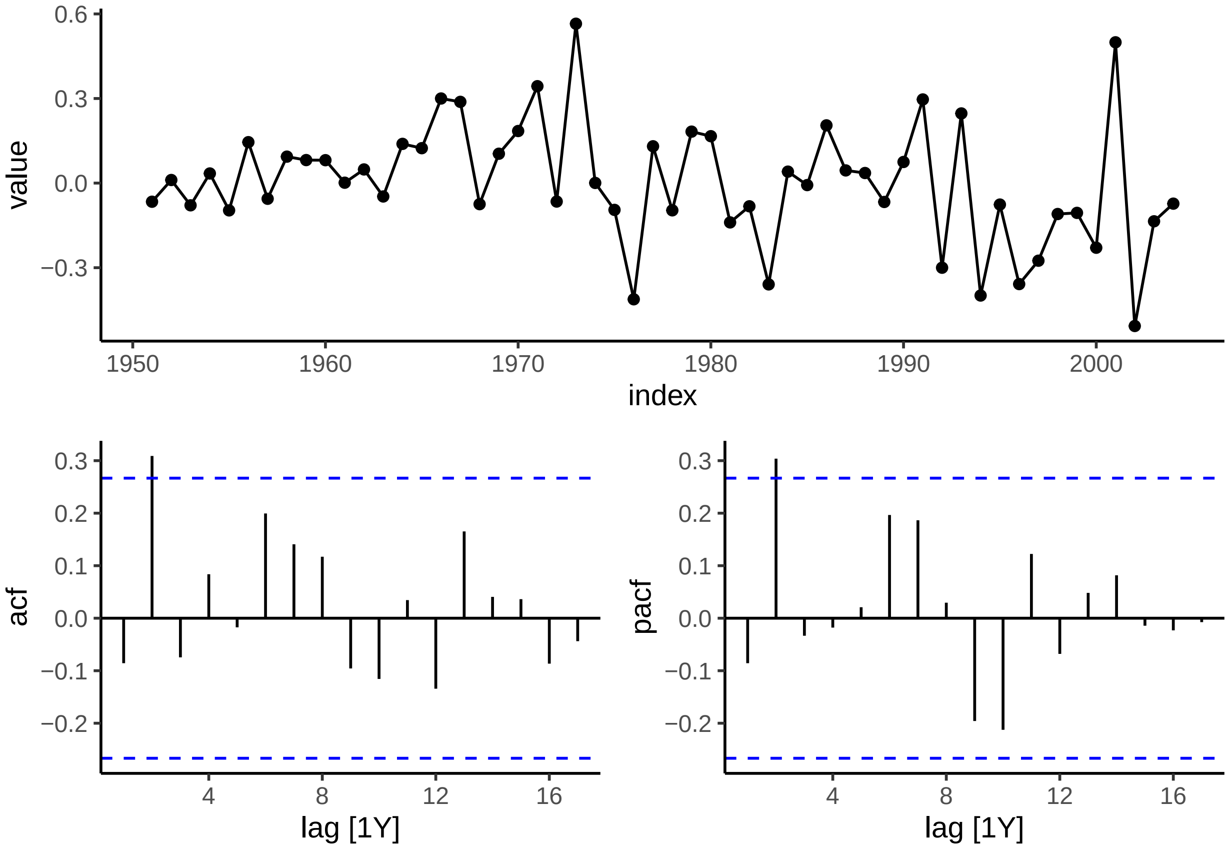
```
wmurders %>% as_tsibble(wmurders) %>%
 gg_tsdisplay(value, plot_type = 'partial')
```



Using only the first difference as instructed to make the data stationary i.e., d = 1

```
wmurders %>% diff() %>% as_tsibble %>%
  gg_tsdisplay(plot_type = "partial")
```

```
## Plot variable not specified, automatically selected ‘y = value‘
```



The results look better as the series seems to get stationary. From the two plots there are significant spikes at lag 2 in ACF and lag 2 in PACF. No spikes beyond lag 2.

The values of p = 2 and q = 2 and the model can be either with p or q having its value as 2

Thus, the data can be modelled by ARIMA with two options: 1. ARIMA(p, 1, 0) = (2, 1, 0) 2. ARIMA(0, 1, q) = (0, 1, 2) There are two equally likely candidate models

Let us choose the model with a Moving Average process(MA) as per instructed.

ARIMA model = ARIMA(0, 1, 2)

```r
ndiffs(wmurders)
```

```
## [1] 2
```

Also "ndiffs" function shows that the data need 2 differencing to make it complete stationary. Let us use d=1 for this assignment.

3) Should you include a constant term in the model? Explain your answer.

No. We should not include a constant term in the model because there is no significant drift in the plots as shown before.

When d = 1, if we include the constant term, the model assumes to include the drift in the series. As per our disscusion, we do not see any drift in the series with diff = 1 and including constant term without drift in the model is wrong.

4) Fitting the model

Let us fit the model with pdq(0, 1, 2)

```
fit <- wmurders %>% as_tsibble() %>%
  model(arima = ARIMA(value ~ pdq(0, 1, 2)))
report(fit)
```

```
## Series: value
## Model: ARIMA(0,1,2)
##
## Coefficients:
##           ma1    ma2
##        -0.066  0.371
## s.e.    0.126  0.164
##
## sigma^2 estimated as 0.0422:  log likelihood=9.71
## AIC=-13.4   AICc=-12.9   BIC=-7.46
```

Let us examine the residuals:

The null hypothesis of the Box Ljung Test: H0 - Model does not show lack of fit(or in simple terms—the model is just fine) The alternate hypothesis: Ha - Model does show a lack of fit

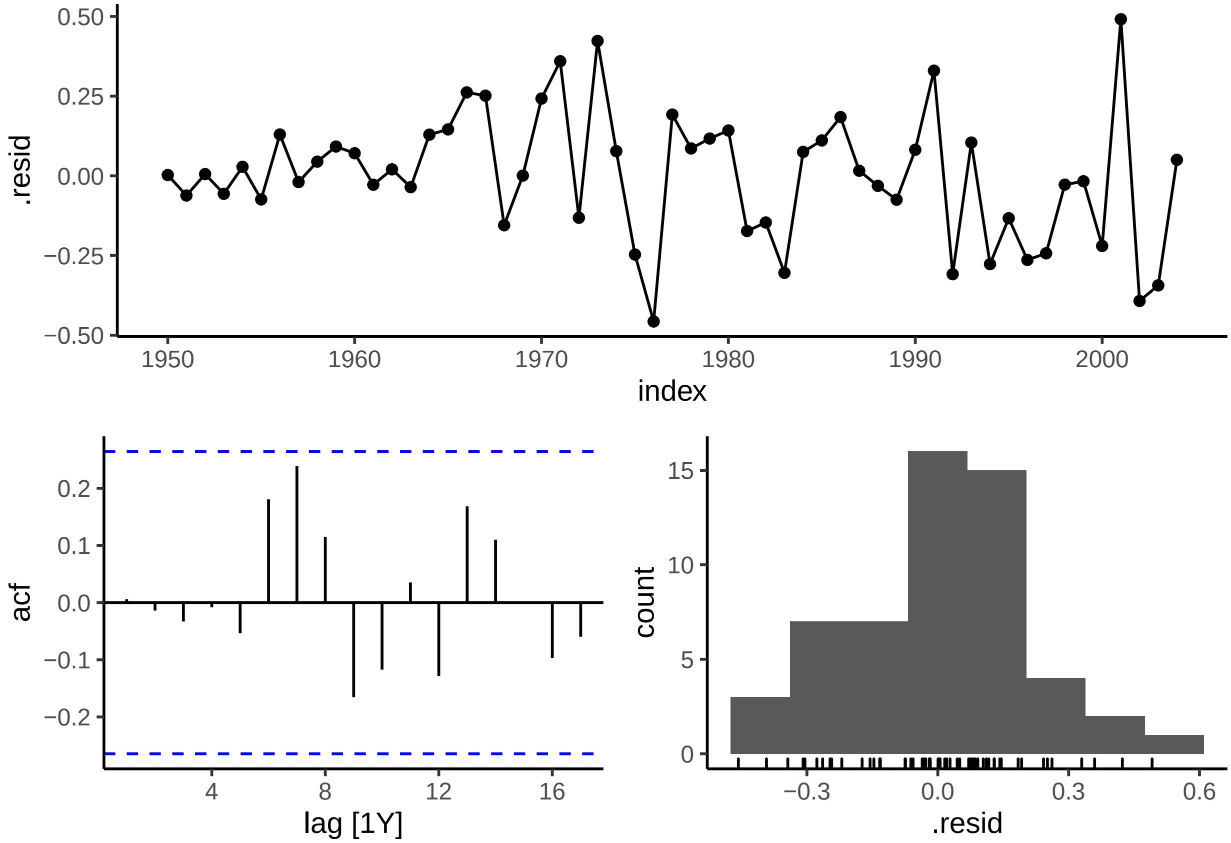
A significant p-value = 0.778 in this test rejects the null hypothesis that the time series isn't autocorrelated.

```
augment(fit) %>%
features(.resid, ljung_box, lag = 3, dof = 2)
```

```
## # A tibble: 1 x 3
##   .model lb_stat lb_pvalue
##   <chr>    <dbl>     <dbl>
## 1 arima   0.0798     0.778
```

Residual plot:

```
gg_tsresiduals(fit)
```



Yes, the model is satisfactory as from the ACF plot of residuals we can observe that all the lags are within the boundaries i.e, no significant autocorrelation in residuals. The residuals of the model can be thought of as a white noise series. But the residuals could be more normally distributed.

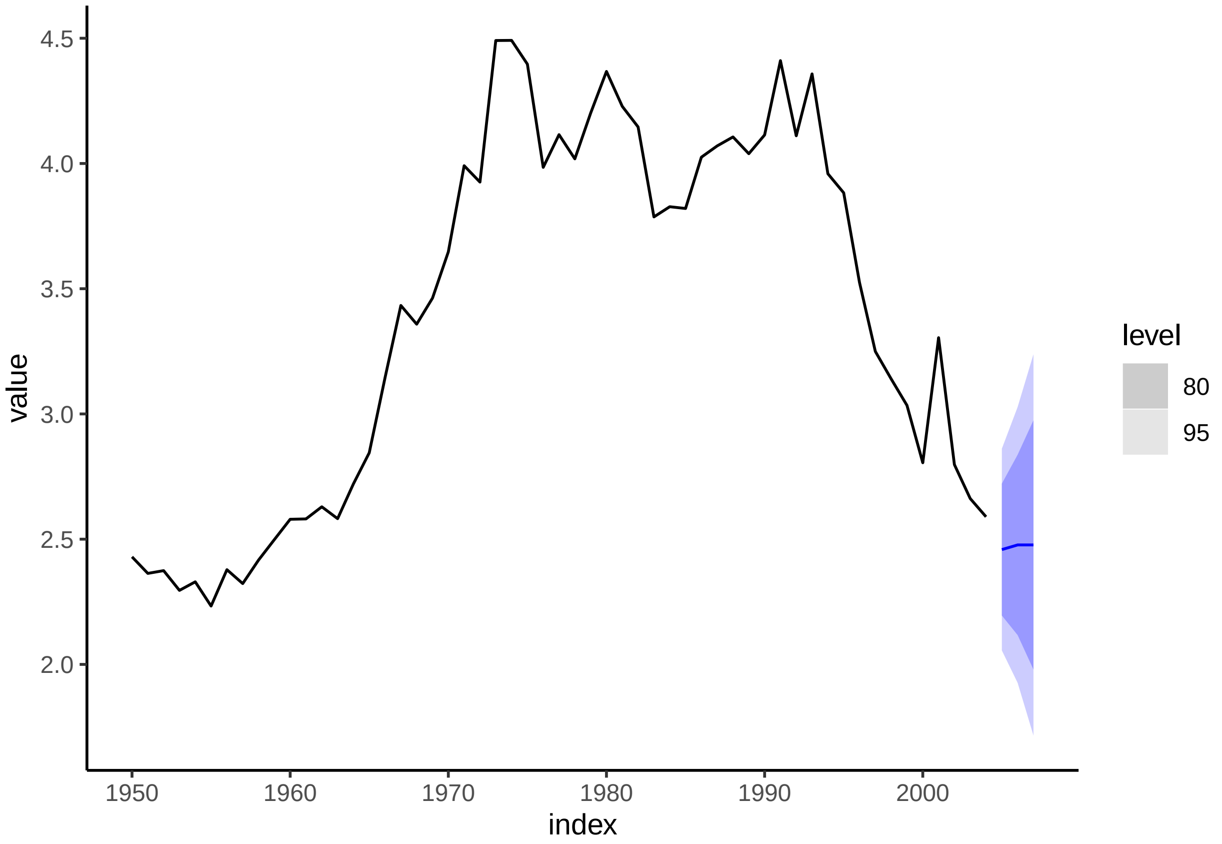
5) Forecasting 3 times ahead with results in a table:

```
fit %>% forecast(h = 3) %>% print.data.frame()
```

```
##   .model index           value .mean
## 1  arima  2005 N(2.46, 0.0422)  2.46
## 2  arima  2006  N(2.48, 0.079)  2.48
## 3  arima  2007  N(2.48, 0.151)  2.48
```

Ploting the series with forecasts and prediction intervals for the next three periods.

```
fit %>% forecast(h = 3) %>% autoplot(wmurders)
```



6) Fitting the model with ARIMA():

1. Let ARIMA () choose the model for d = 1

```
fit.arima = wmurders %>% as_tsibble() %>% model(ARIMA(value ~ pdq(d=1))) %>% report()
```

```
## Series: value
## Model: ARIMA(0,1,0)
##
## sigma^2 estimated as 0.04563:  log likelihood=6.73
## AIC=-11.5   AICc=-11.4   BIC=-9.47
```

2. Force run all the combinations for d = 1

```
fit.arima.all = wmurders %>% as_tsibble()%>% model(ARIMA(value ~ pdq(d=1), stepwise=FALSE, approximation
```

```
## Series: value
## Model: ARIMA(0,1,2)
##
## Coefficients:
##          ma1    ma2
##       -0.066  0.371
## s.e.   0.126  0.164
```

```
##
## sigma^2 estimated as 0.0422:  log likelihood=9.71
## AIC=-13.4   AICc=-12.9   BIC=-7.46
```

Let us compare the three models with Diff = 1

1. My model - fit : AICc = -12.9 pdq(0, 1, 2)
2. ARIMA(model) - fit.arima : AICc = -11.4 pdq(0, 1, 0)
3. All combinations - fit.arima.all : AICc = -12.9 pdq(0, 1, 2)

From the above results we can distinugish the three models according to the AICc values. “fit” and “fit.arima.all” provides the same model with AICc = -12.9. According to the results, “My model Fit” is better than “ARIMA(model) Fit.arima” as the AICc value of “My model Fit” is less compared to AICc value of “ARIMA(model) Fit.arima”.

End of the Assignment