Time Series Analysis of Daily Total Female Births Dataset From Kaggle

Bett

2025-12-21

```r
library(forecast)
```

```
## Registered S3 method overwritten by 'quantmod':
##   method            from
##   as.zoo.data.frame zoo
```

```r
library(tseries)
```

1. Load the data Make sure you've downloaded 'DailyTotalFemaleBirths.csv' from Kaggle and saved in working directory.

```r
data <- read.csv("DailyTotalFemaleBirth.csv")
```

2. Inspect the data structure

```r
str(data)
```

```
## 'data.frame':    365 obs. of  2 variables:
##  $ Date  : chr  "1959-01-01" "1959-01-02" "1959-01-03" "1959-01-04" ...
##  $ Births: int  35 32 30 31 44 29 45 43 38 27 ...
```

```r
head(data)
```

```
##         Date Births
## 1 1959-01-01     35
## 2 1959-01-02     32
## 3 1959-01-03     30
## 4 1959-01-04     31
## 5 1959-01-05     44
## 6 1959-01-06     29
```

The dataset typically has columns: 'Date' and 'Births'. Convert the 'Date' column to Date class

```r
data$Date <- as.Date(data$Date, format="%Y-%m-%d")
```

Create a time series object Since these are daily births for 1959, frequency = 365 is often used for daily data (non-leap year)

```r
births_ts <- ts(data$Births, start=c(1959,1), frequency=365)
```

2. Test for stationarity We can use the Augmented Dickey-Fuller (ADF) test:

```r
adf_result <- adf.test(births_ts)
```

```
## Warning in adf.test(births_ts): p-value smaller than printed p-value
```

```
adf_result
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  births_ts
## Dickey-Fuller = -5.1042, Lag order = 7, p-value = 0.01
## alternative hypothesis: stationary
```

Interpretation:

If the p-value is small (<0.05), we can reject the null hypothesis of non-stationarity. In this dataset, the daily female births data is often already stationary or nearly so. If not stationary, we would consider differencing: *births_ts_diff <- diff(births_ts)*

(Check stationarity on differenced series if needed) *adf.test(births_ts_diff)*

3. Split the data into training and testing sets for model evaluation Let's hold out the last 30 days as test data.

```
train_length <- length(births_ts) - 30
train_ts <- window(births_ts, end=c(1959,(train_length/365)*365))
test_ts <- window(births_ts, start=c(1959, ((train_length+1)/365)*365))
```

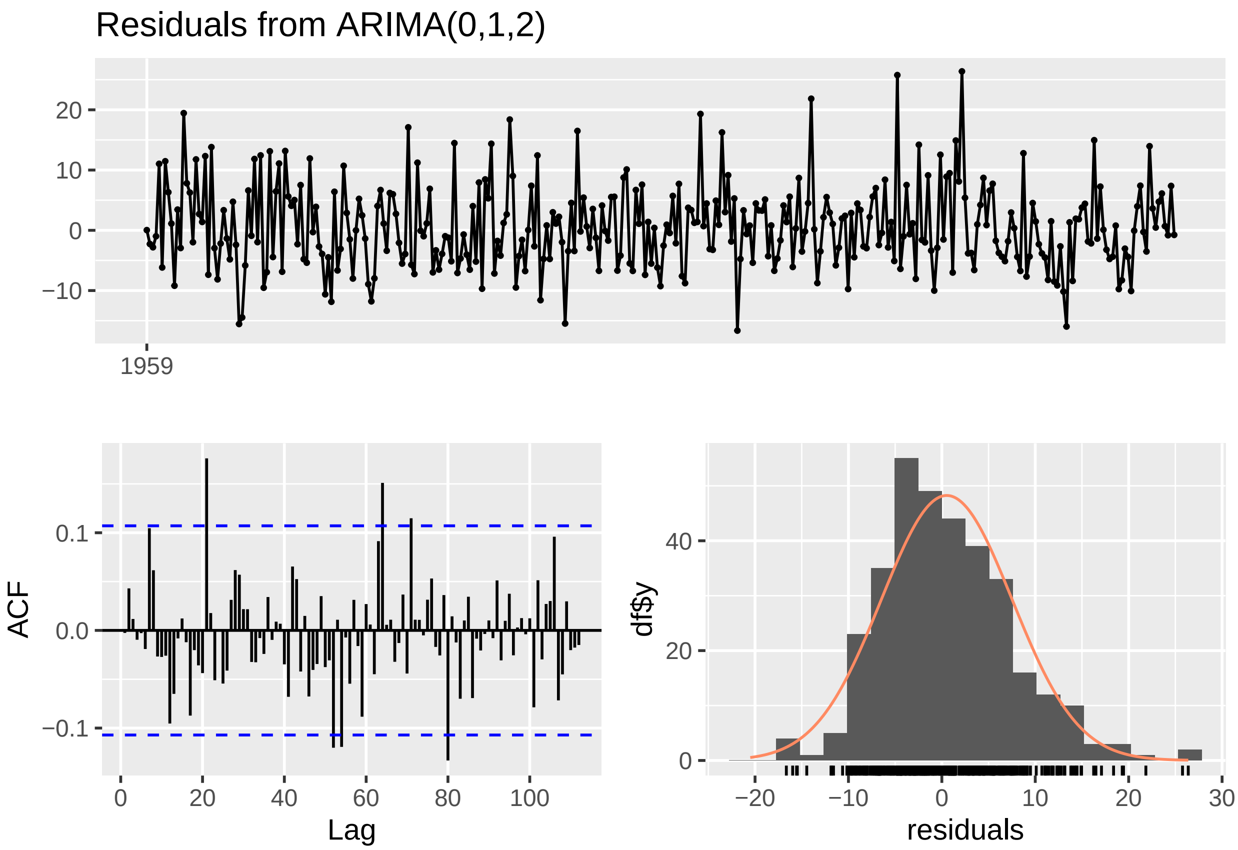
4. Fit an ARIMA model using auto.arima on the training set

```
fit <- auto.arima(train_ts)
summary(fit)
```

```
## Series: train_ts
## ARIMA(0,1,2)
##
## Coefficients:
##           ma1      ma2
##       -0.8573  -0.0964
## s.e.   0.0523   0.0525
##
## sigma^2 = 49.73:  log likelihood = -1126.5
## AIC=2259   AICc=2259.08   BIC=2270.44
##
## Training set error measures:
##                     ME     RMSE      MAE       MPE     MAPE MASE         ACF1
## Training set 0.5224927 7.020607 5.465459 -1.449279 13.26883  NaN -0.002565009
```

Check the residuals of the fitted model

```
checkresiduals(fit)
```

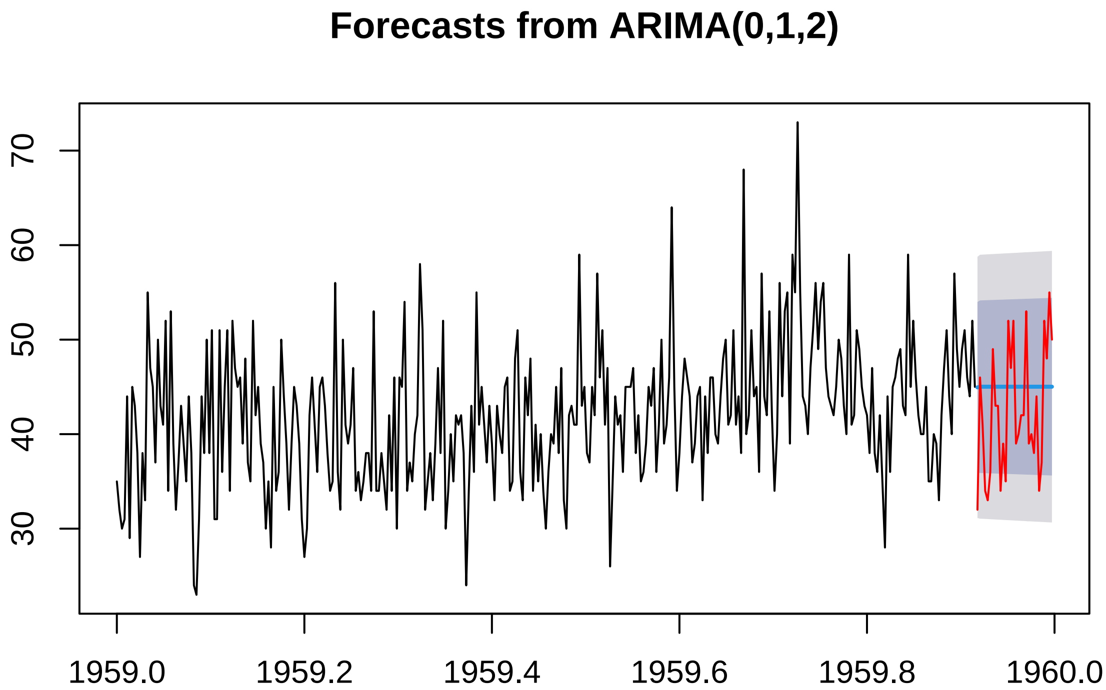



```
##
##  Ljung-Box test
##
## data:  Residuals from ARIMA(0,1,2)
## Q* = 75.269, df = 65, p-value = 0.1801
##
## Model df: 2.   Total lags used: 67
```

If residuals look like white noise (no patterns, no autocorrelations), the model is adequate.

5. Forecast on the test period Forecast for the length of the test set (30 days)

```
fcast <- forecast(fit, h=30)
plot(fcast)
lines(test_ts, col="red", type="l")  # Add the actual test data in red
```



Evaluate forecasting accuracy

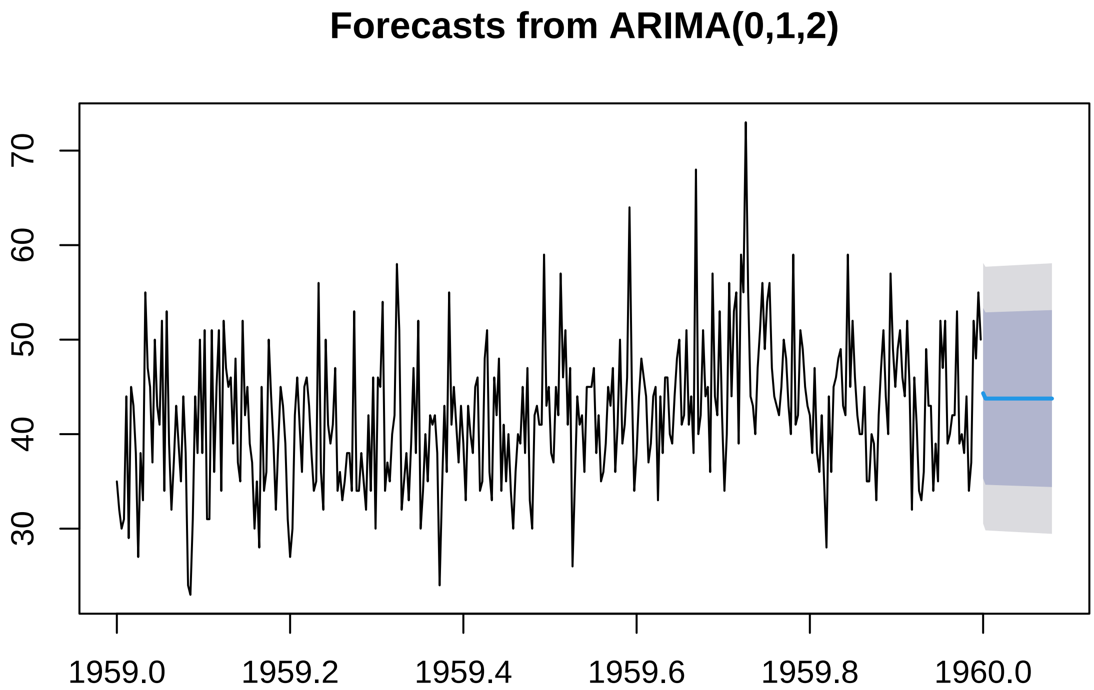
```
accuracy_metrics <- accuracy(fcast, test_ts)
accuracy_metrics
```

```
##                      ME     RMSE      MAE       MPE     MAPE MASE         ACF1
## Training set  0.5224927 7.020607 5.465459 -1.449279 13.26883  NaN -0.002565009
## Test set     -2.7173817 7.160799 6.304155 -9.038007 15.97857  NaN  0.216342192
##              Theil's U
## Training set        NA
## Test set     0.8474797
```

Look at RMSE, MAE, MAPE, etc. to assess forecast accuracy.

6. If satisfied with the model, you can refit using the entire dataset and forecast future values

```
final_fit <- auto.arima(births_ts)
fcast_future <- forecast(final_fit, h=30) # forecasting next 30 days beyond 1959
plot(fcast_future)
```


Forecasts from ARIMA(0,1,2)
70
60
50
40
30
1959.0
1959.2
1959.4
1959.6
1959.8
1960.0