OPSM Assignment

Nirav Mahnot

2023-03-31

#Loading Libraries

```
library(mice)
```

```
## Warning: package 'mice' was built under R version 4.2.3
```

```
##
## Attaching package: 'mice'
```

```
## The following object is masked from 'package:stats':
##
##     filter
```

```
## The following objects are masked from 'package:base':
##
##     cbind, rbind
```

```
library(Amelia)
```

```
## Warning: package 'Amelia' was built under R version 4.2.3
```

```
## Loading required package: Rcpp
```

```
## ##
## ## Amelia II: Multiple Imputation
## ## (Version 1.8.1, built: 2022-11-18)
## ## Copyright (C) 2005-2023 James Honaker, Gary King and Matthew Blackwell
## ## Refer to http://gking.harvard.edu/amelia/ for more information
## ##
```

```
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 4.2.2
```

```
library(forecast)
```

```
## Warning: package 'forecast' was built under R version 4.2.3
```

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

```
library(tseries)
```

```
## Warning: package 'tseries' was built under R version 4.2.3
```

```
library(readxl)
```

```
## Warning: package 'readxl' was built under R version 4.2.3
```

```
library(VIM)
```

```
## Warning: package 'VIM' was built under R version 4.2.3
```

```
## Loading required package: colorspace
```

```
## Loading required package: grid
```

```
## VIM is ready to use.
```

```
## Suggestions and bug-reports can be submitted at: https://github.com/statistikat/VIM/issues
```

```
##
## Attaching package: 'VIM'
```

```
## The following object is masked from 'package:datasets':
##
##     sleep
```

#Loading dataset

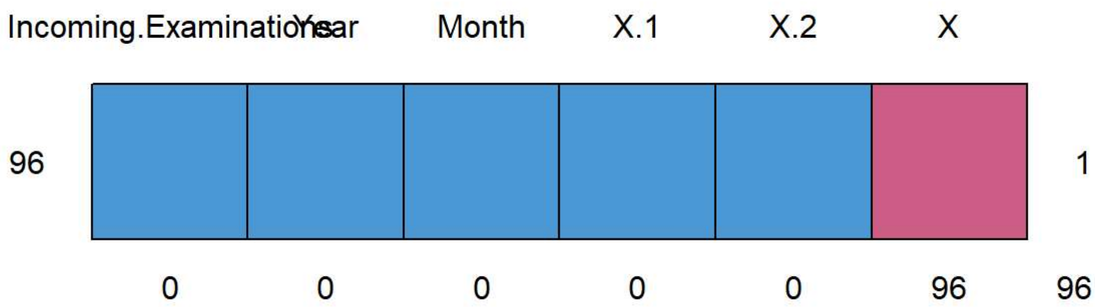
```
getwd()
```

```
## [1] "C:/Users/Nirav Mahnot/Desktop"
```

```
setwd("C:/Users/Nirav Mahnot/Desktop/Ultimately Useless/OPsm324- B.A of analytics/ydiskolaveri")
getwd()
```

```
## [1] "C:/Users/Nirav Mahnot/Desktop/Ultimately Useless/OPsm324- B.A of analytics/ydiskolaveri"
```

```
data <- read.csv("Final BA.csv")
md.pattern(data)
```




```
##    Incoming.Examinations Year Month X.1 X.2  X
## 96                     1    1     1   1   1  0  1
##                        0    0     0   0   0 96 96
```

#MICE

```
imputed_data <- mice(data, m=10, maxit=10, seed=500, print=F)
```

```
## Warning: Number of logged events: 3
```

```
mice_fit <- with(imputed_data, lm(Incoming.Examinations ~ Year + Month))
mice_pooled_fit <- pool(mice_fit)
pool.r.squared(mice_fit, adjusted = TRUE)
```

```
##                est     lo 95     hi 95          fmi
## adj R^2 0.8403302 0.7699623 0.8906402 2.222222e-09
```

```
summary(mice_pooled_fit)
```

```
##          term      estimate   std.error  statistic       df      p.value
## 1 (Intercept) -1.206115e+06 54611.53124 -22.085354 91.05338 2.445273e-38
## 2        Year  6.009950e+02    27.17660  22.114434 91.05338 2.209954e-38
## 3       Month  6.486233e+01    18.03837   3.595797 91.05338 5.249589e-04
```

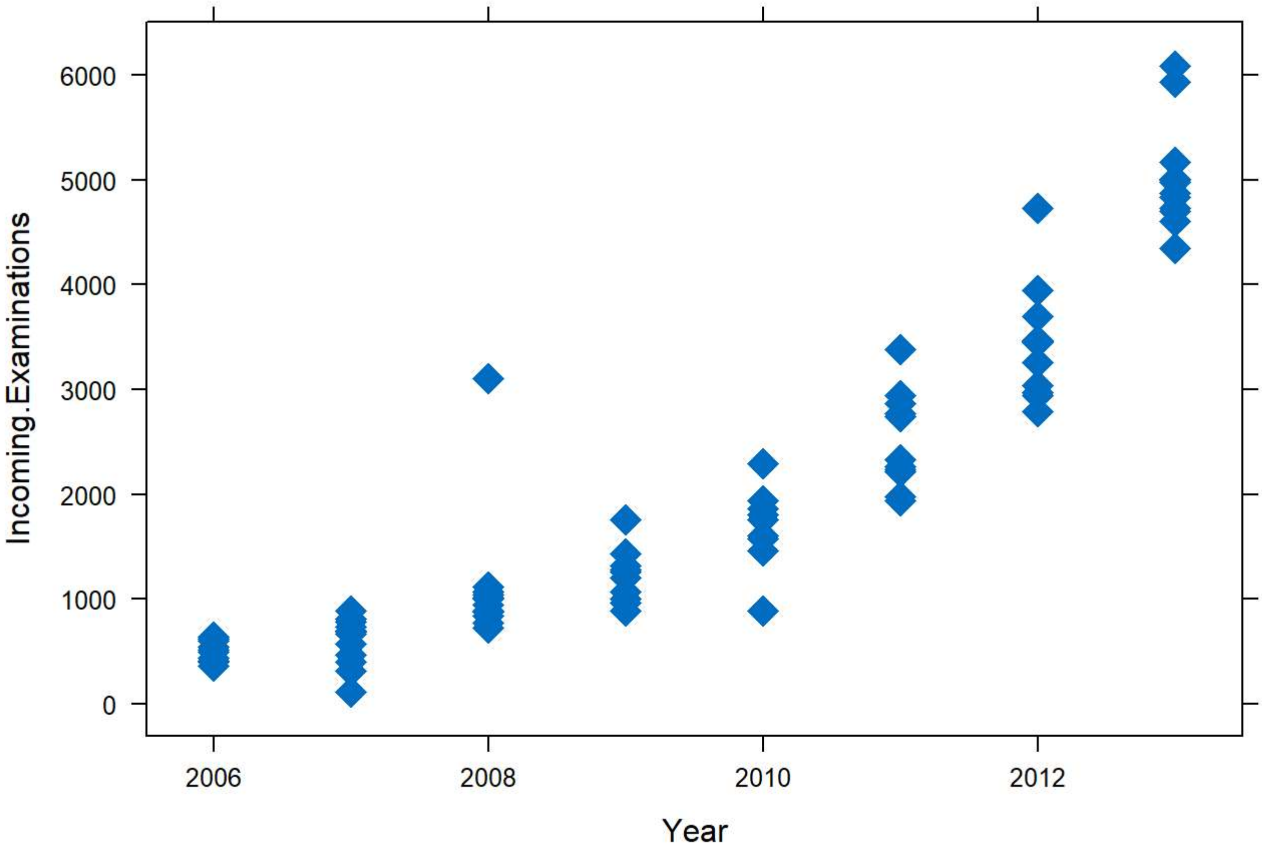
```
completedDataMice <- complete(imputed_data)
completedDataMice
```

```
##    Incoming.Examinations Year Month  X  X.1   X.2
## 1                   4610 2013     1 NA
## 2                   4841 2013     2 NA
## 3                   5172 2013     3 NA
## 4                   4351 2013     4 NA
## 5                   4730 2013     5 NA
## 6                   4706 2013     6 NA
## 7                   5000 2013     7 NA
## 8                   4978 2013     8 NA
## 9                   5008 2013     9 NA
## 10                  6094 2013    10 NA
## 11                  4874 2013    11 NA
## 12                  5933 2013    12 NA
## 13                  2789 2012     1 NA
## 14                  3455 2012     2 NA
## 15                  2940 2012     3 NA
## 16                  2968 2012     4 NA
## 17                  3466 2012     5 NA
## 18                  3037 2012     6 NA
## 19                  3946 2012     7 NA
## 20                  3459 2012     8 NA
## 21                  3446 2012     9 NA
## 22                  3258 2012    10 NA
## 23                  4729 2012    11 NA
## 24                  3694 2012    12 NA
## 25                  1934 2011     1 NA MICE NOICE
## 26                  2334 2011     2 NA
## 27                  1973 2011     3 NA
## 28                  2262 2011     4 NA
## 29                  2259 2011     5 NA
## 30                  2217 2011     6 NA
## 31                  2739 2011     7 NA
## 32                  2772 2011     8 NA
## 33                  3383 2011     9 NA
## 34                  2869 2011    10 NA
## 35                  2239 2011    11 NA
## 36                  2940 2011    12 NA MICE
## 37                   890 2010     1 NA MICE
## 38                  1578 2010     2 NA MICE
## 39                  1578 2010     3 NA
## 40                  1604 2010     4 NA
## 41                  1758 2010     5 NA
## 42                  1457 2010     6 NA MICE
## 43                  1457 2010     7 NA
## 44                  1607 2010     8 NA
## 45                  1808 2010     9 NA
## 46                  1866 2010    10 NA
## 47                  1934 2010    11 NA
## 48                  2294 2010    12 NA
## 49                  1004 2009     1 NA
## 50                  1065 2009     2 NA
## 51                  1263 2009     3 NA
```

```
## 52                   962 2009     4 NA
## 53                  1065 2009     5 NA
## 54                  1429 2009     6 NA
## 55                  1205 2009     7 NA
## 56                   890 2009     8 NA
## 57                  1320 2009     9 NA
## 58                  1276 2009    10 NA
## 59                  1757 2009    11 NA
## 60                  1757 2009    12 NA
## 61                   875 2008     1 NA
## 62                   840 2008     2 NA
## 63                   724 2008     3 NA
## 64                  1115 2008     4 NA
## 65                   997 2008     5 NA
## 66                   775 2008     6 NA
## 67                   886 2008     7 NA
## 68                  1041 2008     8 NA
## 69                  1011 2008     9 NA
## 70                  3110 2008    10 NA
## 71                   939 2008    11 NA
## 72                  1065 2008    12 NA
## 73                   398 2007     1 NA
## 74                   311 2007     2 NA
## 75                   664 2007     3 NA
## 76                   680 2007     4 NA
## 77                   107 2007     5 NA
## 78                   467 2007     6 NA
## 79                   566 2007     7 NA
## 80                   806 2007     8 NA
## 81                   732 2007     9 NA
## 82                   886 2007    10 NA
## 83                   776 2007    11 NA
## 84                   698 2007    12 NA
## 85                   362 2006     1 NA
## 86                   436 2006     2 NA
## 87                   393 2006     3 NA
## 88                   490 2006     4 NA
## 89                   508 2006     5 NA
## 90                   508 2006     6 NA
## 91                   393 2006     7 NA
## 92                   596 2006     8 NA
## 93                   634 2006     9 NA
## 94                   613 2006    10 NA
## 95                   545 2006    11 NA
## 96                   411 2006    12 NA
```

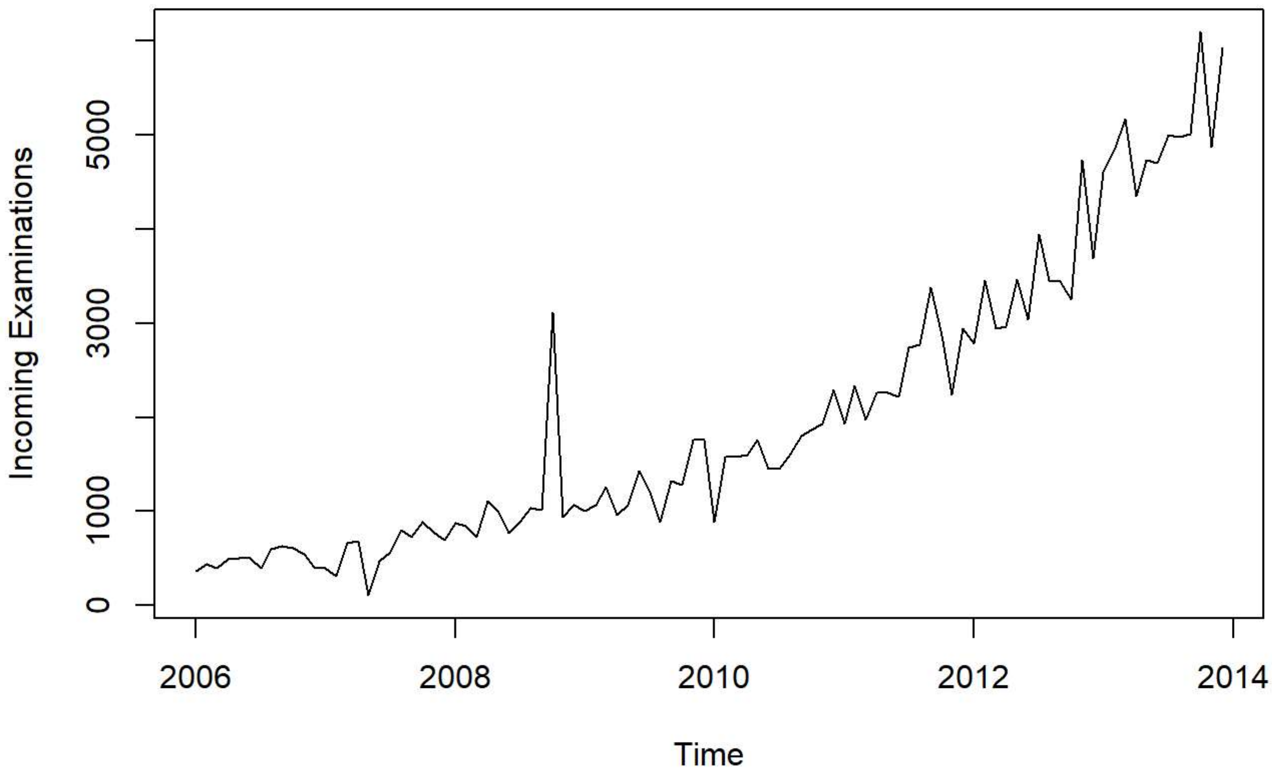
```
write.csv(completedDataMice, "cleaned_dataset.csv", row.names=F)

completedDataMice <- completedDataMice %>%
  dplyr::arrange(Year)
```

```
xyplot(imputed_data, Incoming.Examinations ~ Year, pch=18,cex=2)
```

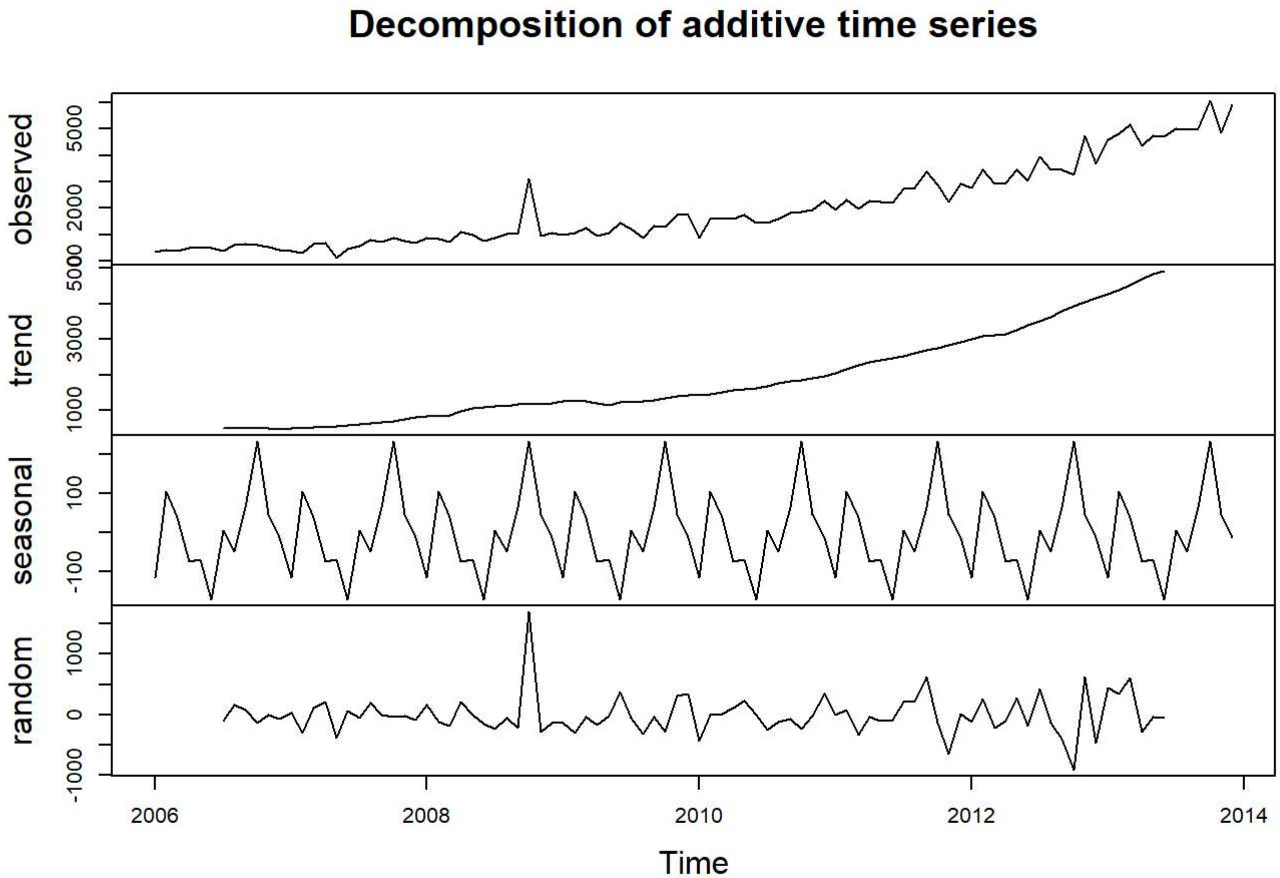



```
# Density plot of the imputed dataset (96 observations)
#densityplot(imputed_data, n=96)
```

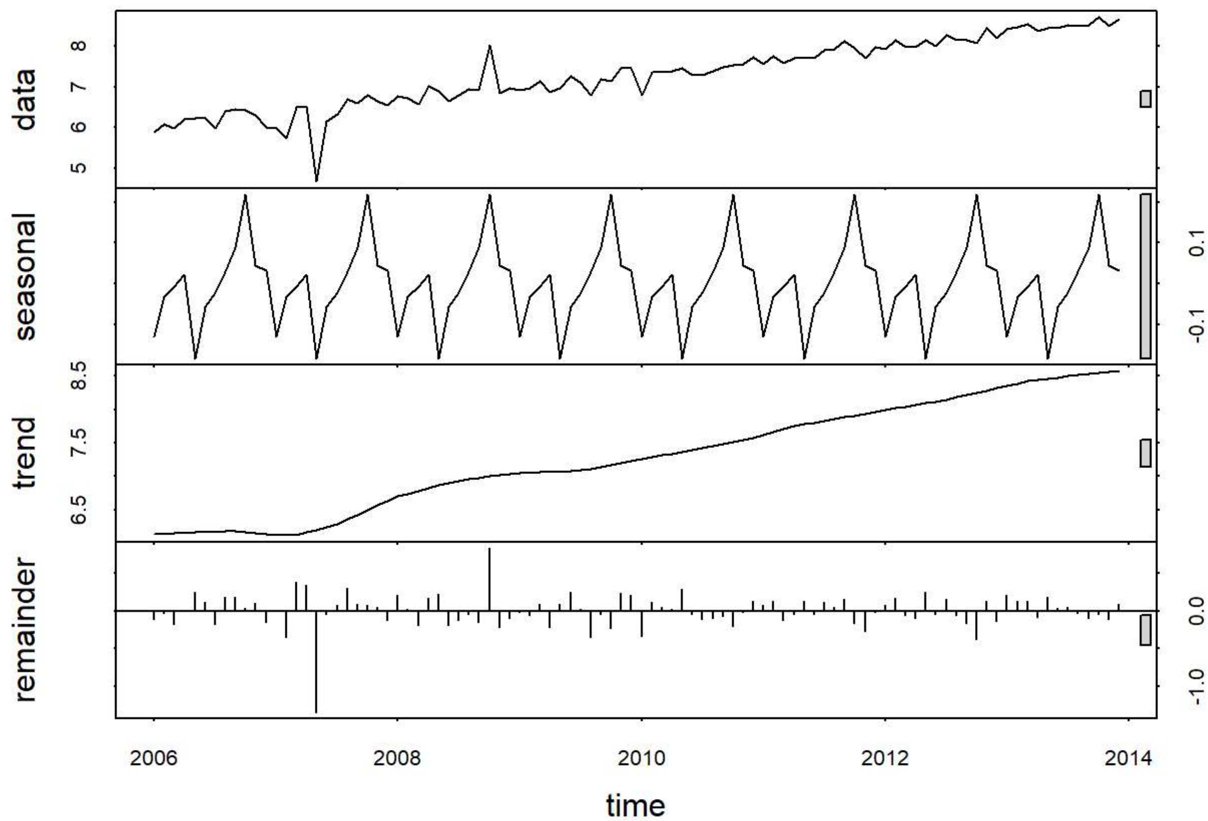
```
# create time series
ts = ts(completedDataMice$Incoming.Examinations, start=c(2006, 1), end=c(2013,12),frequency=12)

# plot the time series
plot(ts, xlab="Time", ylab="Incoming Examinations")
```



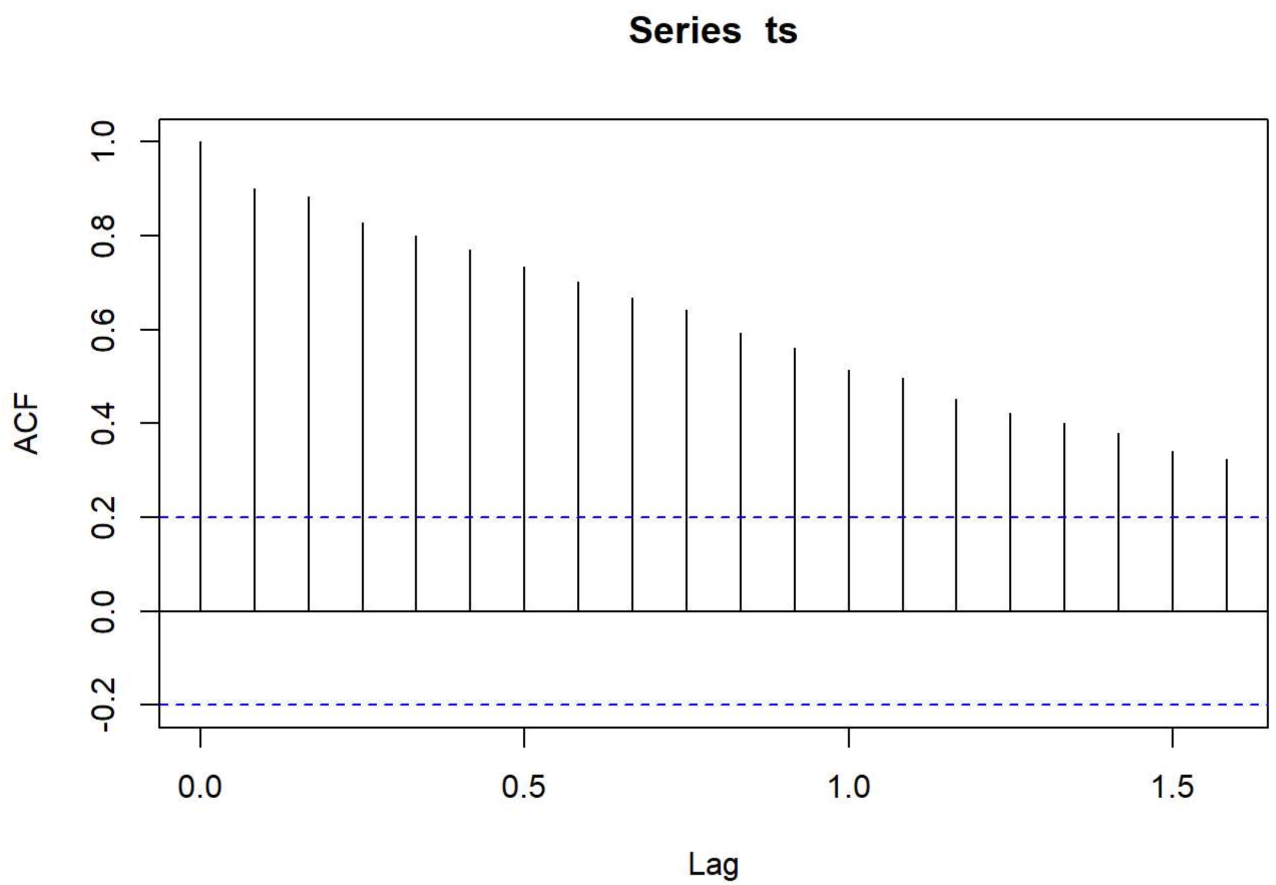

```
# decompose the time series to understand the prevalence of three componenets: trend,seasonalit
y, and error/irregularity
components.ts = decompose(ts)


# we see that the data is not stationary
plot(components.ts)
```




```
# we can test this another way by using seasonal decomposition in loess smoothing
plot(stl(log(ts), s.window="period"))
```




```
# autocorrelation function shows data is not stationary. Correlates set of observations
# at current time to set of observastions at k periods earlier.
acf(ts)
```




```
# run ADF test to confirm our hypothesis that data is not stationary
adf.test(ts)
```

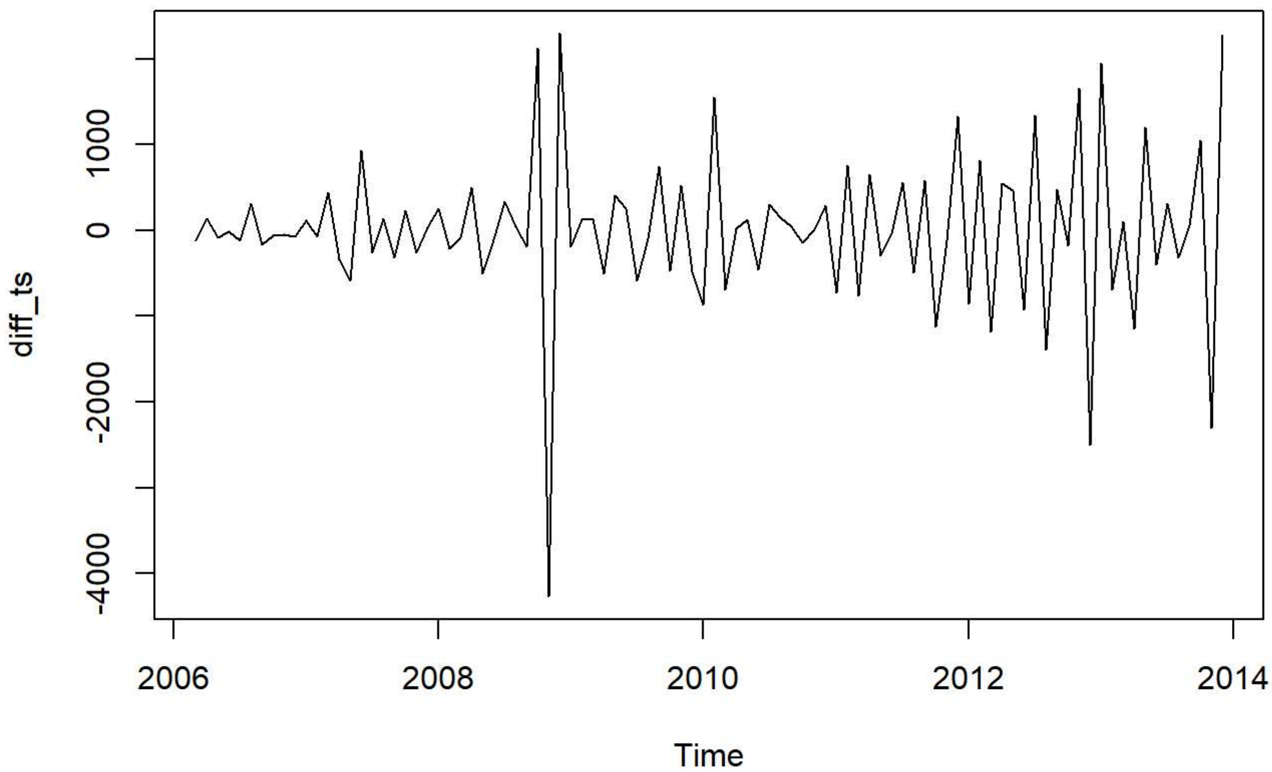
```
##
##  Augmented Dickey-Fuller Test
##
## data:  ts
## Dickey-Fuller = -0.46095, Lag order = 4, p-value = 0.9818
## alternative hypothesis: stationary
```

```
# estimate the number of differences required to make a given time series stationary
ndiffs(ts)
```

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

```
# differentiate the data to make it stationary. We differentiate it twice to remove what appears to be a quadratic trend.
diff_ts <- diff(ts, differences=2)
```

```
# take a look at the differentiated data
plot(diff_ts)
```

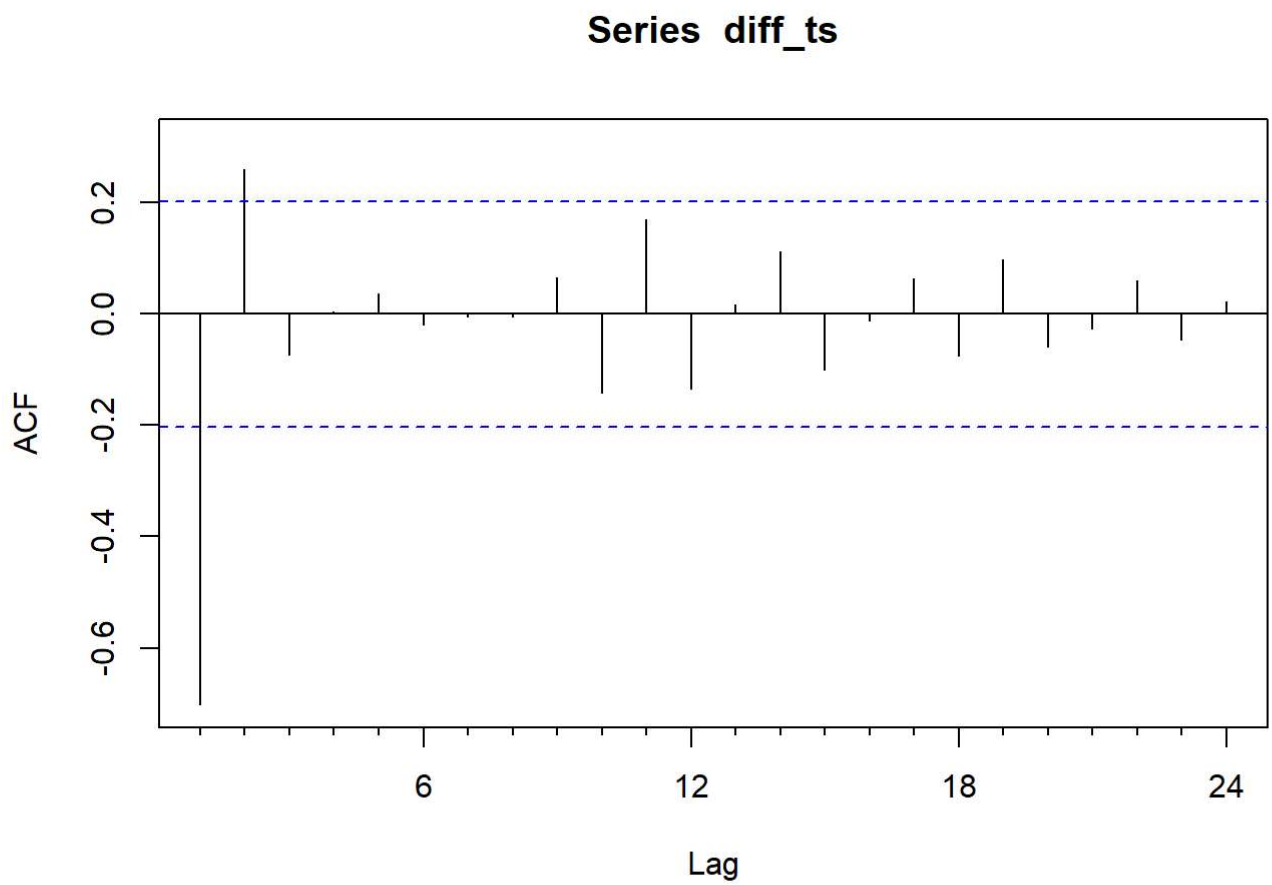



```
# run ADF to confirm data is now stationary
adf.test(diff_ts)
```

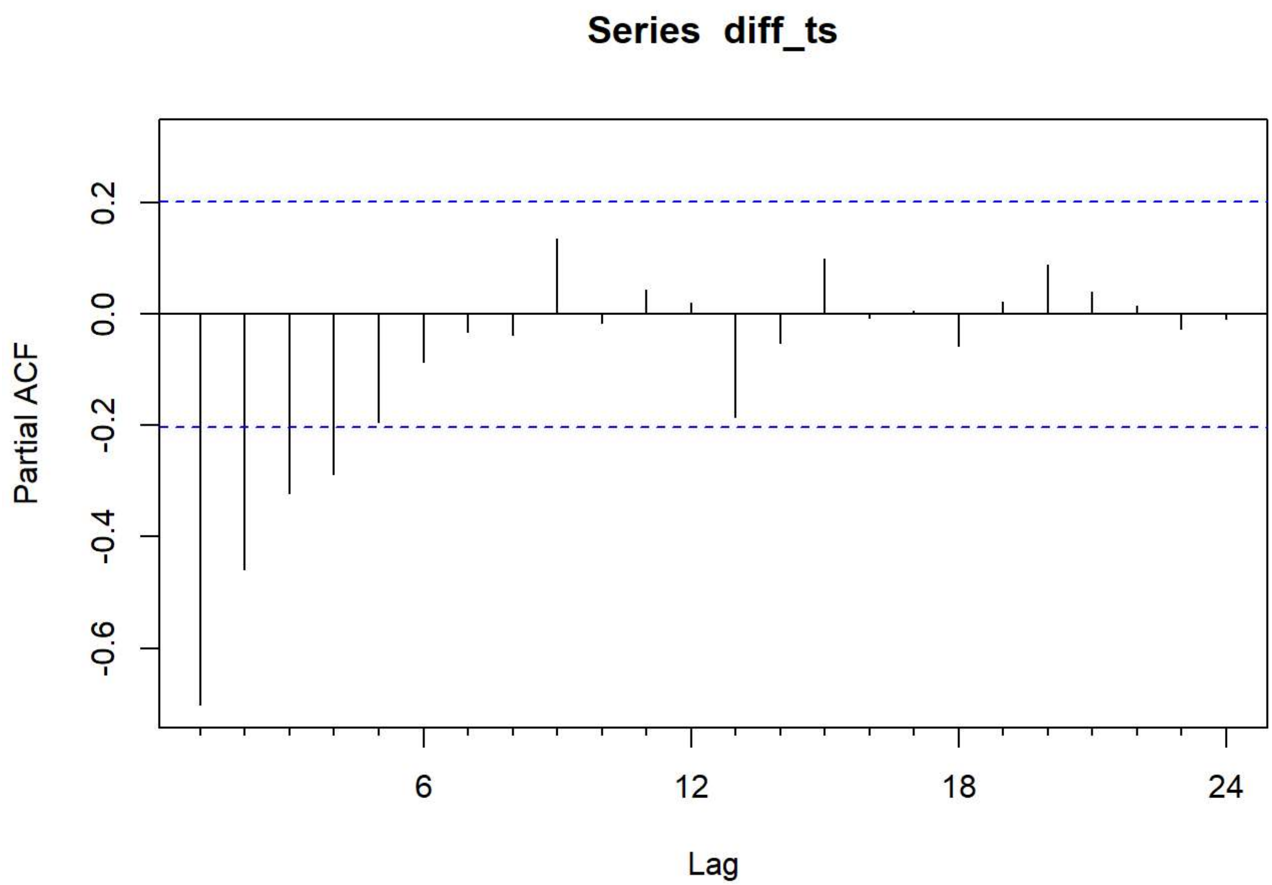
```
## Warning in adf.test(diff_ts): p-value smaller than printed p-value
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  diff_ts
## Dickey-Fuller = -9.4906, Lag order = 4, p-value = 0.01
## alternative hypothesis: stationary
```

```
# review the ACF chart of the differenced time series. This plots the AC at various lags for the stationary/differenced time series.
Acf(diff_ts)
```




```
# correlation between an observation at current period and an observation at k periods earlier w
ith observations between removed.
Pacf(diff_ts)
```




```
# Automated forecasting using an ARIMA model
fitauto <- auto.arima(ts)

summary(fitauto)
```

```
## Series: ts
## ARIMA(1,1,1) with drift
##
## Coefficients:
##           ar1      ma1    drift
##       -0.2096  -0.6164  54.6354
## s.e.   0.1291   0.0967  13.2146
##
## sigma^2 = 163854:  log likelihood = -703.98
## AIC=1415.96   AICc=1416.4   BIC=1426.17
##
## Training set error measures:
##                     ME     RMSE      MAE      MPE     MAPE      MASE
## Training set -1.059456 396.2662 256.6438 -14.6042 23.24845 0.3590261
##                      ACF1
## Training set -0.005927131
```

```
# correlation of custom ARIMA fitted values to actual values
cor(fitted(fitauto),ts)^2
```

```
## [1] 0.9361093
```

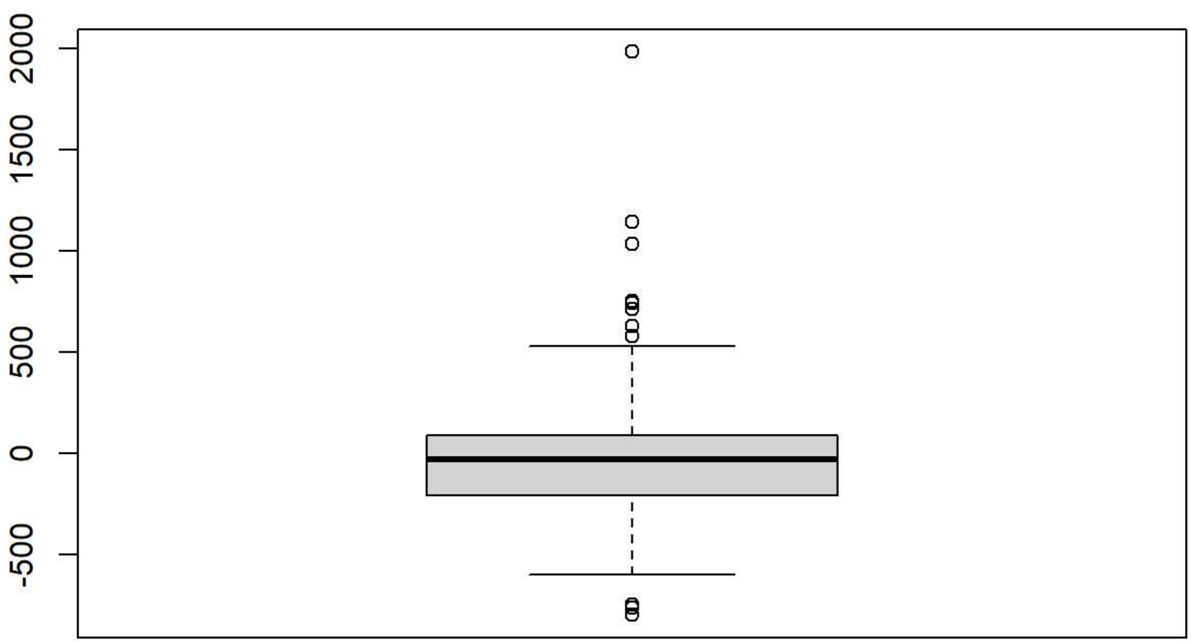
```
accuracy(fitauto) # check accuracy for ARIMA auto fit
```

```
##                     ME     RMSE      MAE      MPE     MAPE      MASE
## Training set -1.059456 396.2662 256.6438 -14.6042 23.24845 0.3590261
##                      ACF1
## Training set -0.005927131
```

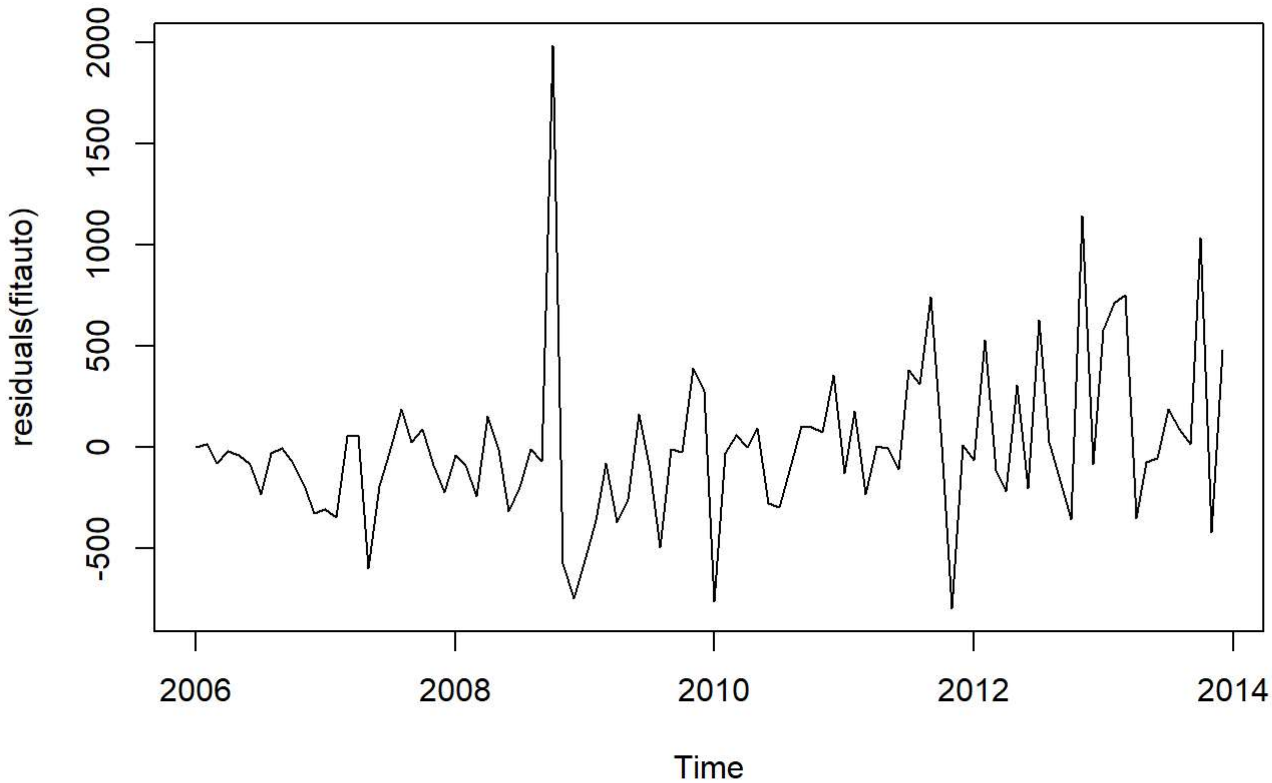
```
# checking residuals

summary(residuals(fitauto))
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## -800.42 -206.91  -30.97   -1.06   89.43 1983.63
```

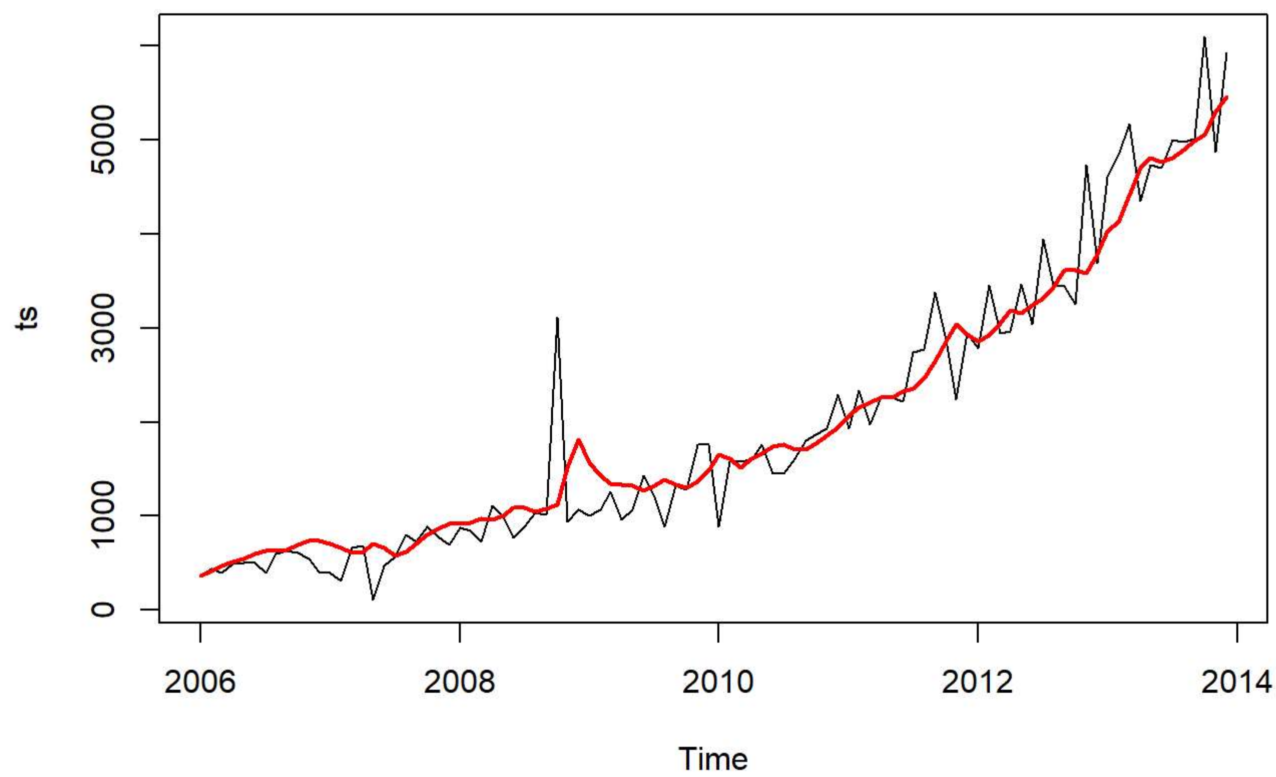
```
boxplot(residuals(fitauto))
```



```
plot(residuals(fitauto))
```

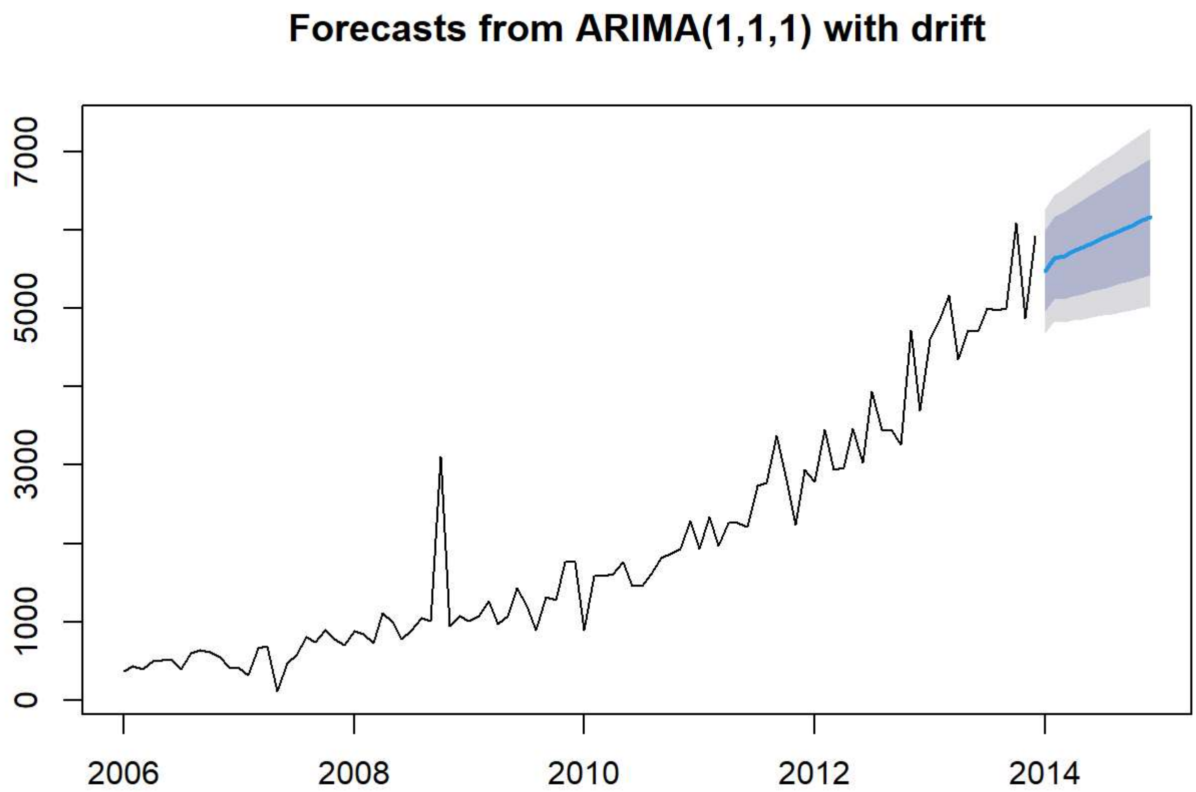



```
# compare ARIMA models to time series
plot(ts)
lines(fitted(fitauto), col="red", lwd="2")
```




```
# create forecast for auto ARIMA model
forecast_arimaauto <- forecast(fitauto, h=12)
```

```
# plot auto generated forecast
plot(forecast_arimaauto)
```




```
# show forecasted incoming exams for next 12 months using the auto ARIMA model
forecast_arimaauto
```

```
##          Point Forecast    Lo 80    Hi 80    Lo 95    Hi 95
## Jan 2014       5482.531 4963.773 6001.289 4689.159 6275.903
## Feb 2014       5643.029 5116.472 6169.585 4837.730 6448.327
## Mar 2014       5675.477 5118.976 6231.978 4824.383 6526.571
## Apr 2014       5734.762 5155.366 6314.158 4848.653 6620.872
## May 2014       5788.423 5185.933 6390.913 4866.994 6709.852
## Jun 2014       5843.263 5218.750 6467.776 4888.153 6798.373
## Jul 2014       5897.856 5252.026 6543.685 4910.145 6885.566
## Aug 2014       5952.500 5286.044 6618.955 4933.244 6971.756
## Sep 2014       6007.133 5320.670 6693.597 4957.278 7056.989
## Oct 2014       6061.769 5355.864 6767.674 4982.181 7141.358
## Nov 2014       6116.405 5391.580 6841.229 5007.881 7224.929
## Dec 2014       6171.040 5427.777 6914.303 5034.317 7307.763
```

#Forecasting with Holt's (with MICE imputed dataset)

```
# generate time series
ts = ts(completedDataMice$Incoming.Examinations, start=c(2006, 1), end=c(2013,12),frequency=12)
```

```
# Holt's approach but with multiplicative error and multiplicative trend.
fitholts_m <- ets(ts, model="MMN")
```

```
# Holt's model (additive)
fitholts_a <- ets(ts, model="AAN")
```

```
# check Holt's multiplicative accuracy.
accuracy(fitholts_m)
```

```
##                    ME     RMSE      MAE       MPE     MAPE      MASE      ACF1
## Training set 32.80243 407.0672 264.4954 -11.81497 22.19381 0.3700098 0.1492489
```

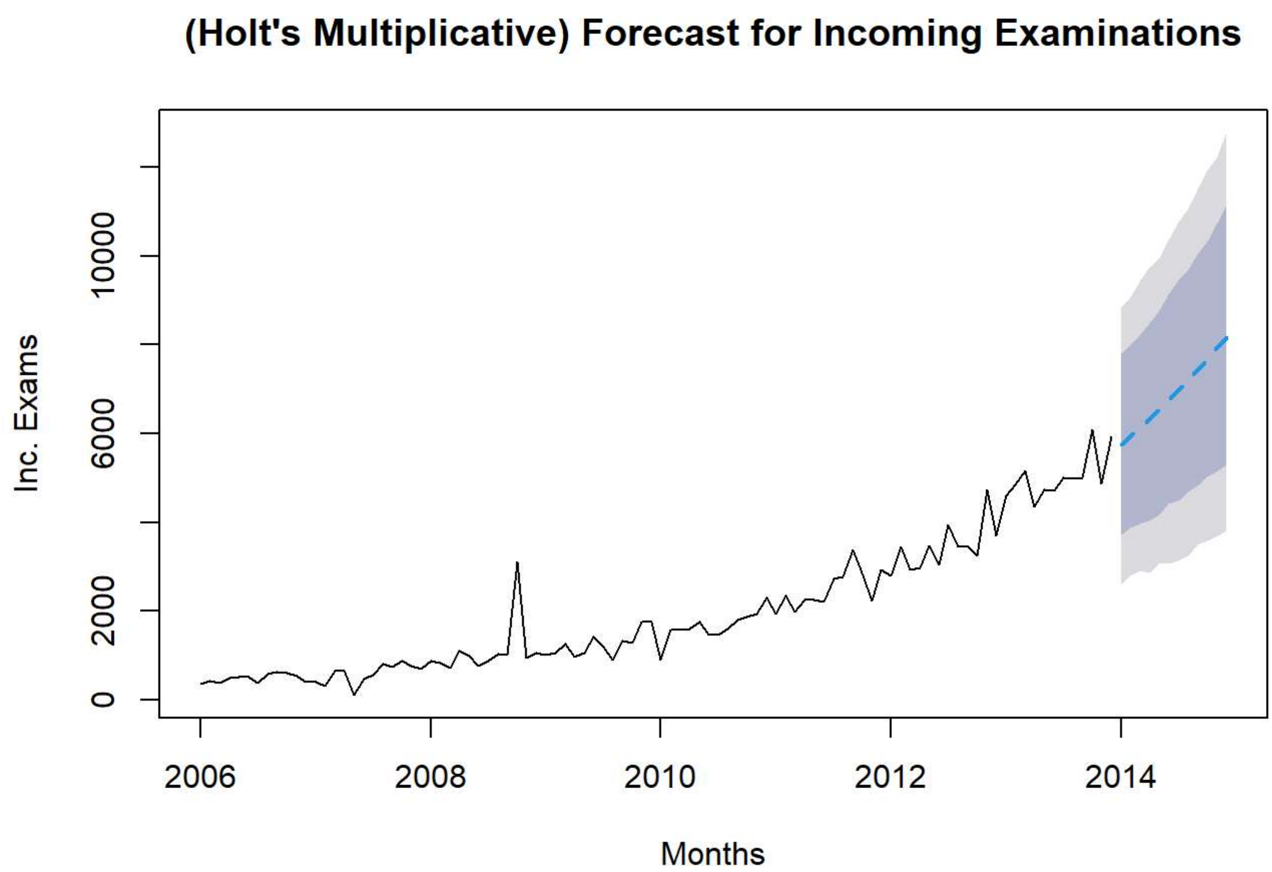
```
# check Holt's additive model accuracy
accuracy(fitholts_a)
```

```
##                   ME     RMSE      MAE       MPE     MAPE      MASE        ACF1
## Training set 60.6548 383.3351 259.3693 -4.553655 20.56221 0.3628389 -0.08718459
```

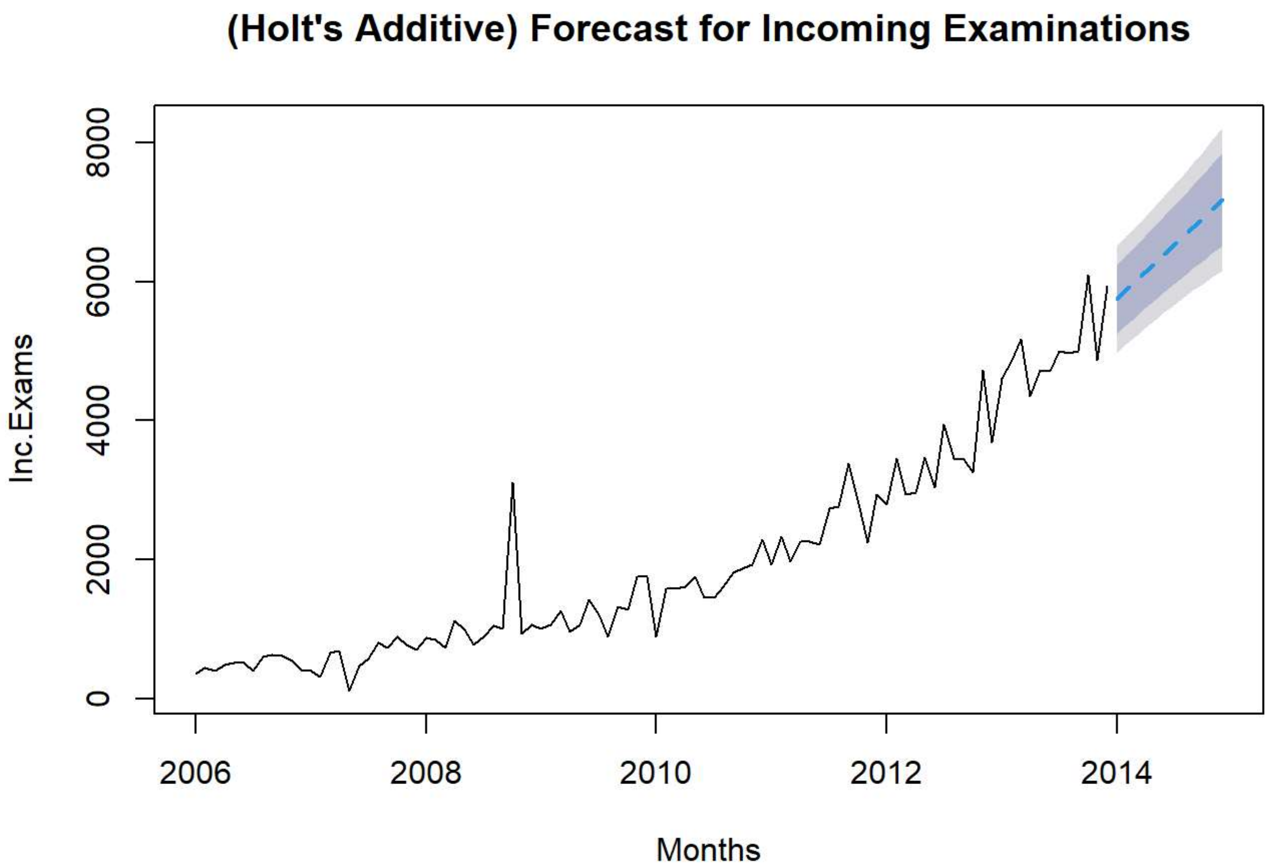
```
# create Holts multiplicative forecast
forecast_m <- forecast(fitholts_m, 12)
```

```
# create Holts additive forecast for comparison
forecast_a <- forecast(fitholts_a, 12)
```

```
# plot Holts multiplicative forecast
plot(forecast_m, main="(Holt's Multiplicative) Forecast for Incoming Examinations",ylab="Inc. Ex
ams", xlab="Months", flty=2)
```




```
# plot Holts forecast
plot(forecast_a, main="(Holt's Additive) Forecast for Incoming Examinations", ylab="Inc.Exams",
xlab="Months", flty=2)
```




```
# show forecasted incoming exams for next 12 months by Holts_m model
forecast_m
```

```
##          Point Forecast    Lo 80     Hi 80    Lo 95     Hi 95
## Jan 2014       5747.502 3722.853  7802.523 2623.163  8851.292
## Feb 2014       5934.660 3887.895  7996.718 2792.182  9070.853
## Mar 2014       6127.912 3962.535  8228.738 2908.320  9445.132
## Apr 2014       6327.458 4045.951  8480.900 2855.054  9749.163
## May 2014       6533.501 4175.469  8776.152 3074.198  9959.735
## Jun 2014       6746.254 4425.775  9127.789 3065.471 10351.574
## Jul 2014       6965.935 4496.113  9458.795 3134.126 10758.241
## Aug 2014       7192.769 4686.414  9701.536 3257.633 11069.503
## Sep 2014       7426.990 4820.380 10049.082 3485.924 11511.102
## Oct 2014       7668.838 5035.754 10342.344 3591.059 11939.407
## Nov 2014       7918.561 5149.100 10712.080 3676.503 12199.618
## Dec 2014       8176.416 5293.368 11139.425 3817.259 12783.936
```

```
# show forecasted incoming exams for next 12 months by Holts_a model
forecast_a
```

```
##          Point Forecast    Lo 80    Hi 80    Lo 95    Hi 95
## Jan 2014       5747.274 5245.444 6249.104 4979.791 6514.756
## Feb 2014       5878.159 5370.489 6385.829 5101.745 6654.574
## Mar 2014       6009.044 5493.895 6524.194 5221.191 6796.898
## Apr 2014       6139.930 5615.514 6664.345 5337.905 6941.954
## May 2014       6270.815 5735.226 6806.403 5451.703 7089.927
## Jun 2014       6401.700 5852.941 6950.460 5562.445 7240.955
## Jul 2014       6532.585 5968.594 7096.577 5670.036 7395.135
## Aug 2014       6663.471 6082.153 7244.788 5774.423 7552.519
## Sep 2014       6794.356 6193.609 7395.103 5875.593 7713.119
## Oct 2014       6925.241 6302.975 7547.508 5973.567 7876.916
## Nov 2014       7056.127 6410.283 7701.970 6068.395 8043.859
## Dec 2014       7187.012 6515.581 7858.443 6160.147 8213.877
```

```
#summaries of Holt’s multiplicative and additive models

summary(fitholts_m$residuals)
```

```
##      Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
## -0.847856 -0.148299 -0.007788 -0.017017  0.102030  1.785697
```

```
summary(fitholts_a$residuals)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## -657.89 -151.41   49.72   60.65  180.27 2081.26
```