Report for DACS Contract

00098260 (Project Number 00124438)

Running head: Breeding bird surveillance

Corresponding author:
Jonathan F. Day
Florida Medical Entomology Lab
200 9th St. SE, Vero Beach,
Florida 32962, USA
Phone: (772) 778-7200 x 132
Fax: (772) 778-7205
E-mail: jfda@ufl.edu

Evaluation of the Association of Long-Term Breeding Bird Surveys and Audubon Christmas Count Data with the Transmission of St. Louis Encephalitis Virus and West Nile Virus to Sentinel Chickens

Jonathan F. Day and Gregory K. Ross
University of Florida, IFAS, Florida Medical Entomology Laboratory
200 9th St. SE, Vero Beach, FL 32962

ABSTRACT

We have known for some time that the relationships and interactions between environmental conditions, virus, mosquito vector, and avian hosts set the stage for Saint Louis encephalitis virus (SLEV) and/or West Nile virus (WNV) amplification and transmission in Florida. Field studies have indicated that Blue Jay, Common Grackle, Mourning Dove, Northern Cardinal, and Northern Mockingbird populations provide the critical amplification link necessary to support SLEV and WNV epidemics. Currently, mosquito control programs in Florida can track the amplification and transmission of arbovirus through the use of sentinel chicken flocks. Along with the estimation of mosquito vector populations and real-time tracking of environmental factors that can trigger epidemics, local mosquito control arbovirus surveillance programs have a host of tools to calculate the risk of arboviral transmission to humans. One tool that is not readily available to mosquito control is the annual avian breeding success and the distribution of fledgling birds. These two important components of avian populations directly impact arboviral amplification, transmission, and the probability of widespread arboviral epidemics. To fill this gap, we investigated the association of Breeding Bird Survey and Christmas Bird Count Data with SLEV transmission to sentinel chickens from 1978 to 2015 across Peninsular Florida using data from 37 counties. Using a modified ARIMA (Box-Jenkins) Modeling Algorithm for Time Series, we compared both BBS average counts per route and trend indexes against SLE sentinel chicken seroconversions on an annual basis. We identified positive correlations between SLE transmission to sentinel chickens and Common Grackle population averages 4 and 5 years prior. Various other significant positive and negative correlations were found between BBS data and SLE transmission to sentinel chickens in prior years for Blue Jay, Common Grackle, Northern Cardinal and Northern Mockingbird species. No correlations were evident for Mourning Dove populations. Further analysis of each significant for Granger causality identified only one significant correlation in which SLE transmission leads Northern Mockingbird average by 9 years. The output of the study has shown that the Common Grackle may be useful as an indicator for increased SLE transmission across Peninsular Florida in following years and requires further study. Further examination of the correlations identified in this study would be beneficial to Florida mosquito control programs by adding to their arboviral surveillance capabilities without added costs to their program.

Key Words

arboviral surveillance, breeding birds, arboviral transmission

INTRODUCTION

The relationships and interactions between environmental conditions, virus, mosquito vector, and avian hosts have been shown to determine the state of Saint Louis encephalitis virus (SLEV) and/or West Nile virus (WNV) amplification and transmission in Florida. In an effort to better understand those relationships and interactions, these four factors have been classified into two groups: Primers and Triggers. Primers are defined as the biotic conditions that align in space and time before major SLEV/WNV epidemics and include the virus, mosquito vectors, and avian hosts. Triggers, which are typically abiotic in nature, would include environmental conditions and patterns. The proper alignment of the primers leads to increase amplification of the virus in both the mosquito vector and avian hosts (wild birds). If the amplification of the virus reaches a threshold and environmental trigger is introduced, an epidemic is possible (Day 2001).

In an effort to predict and prepare for epidemics, many mosquito control programs in Florida run arboviral surveillance programs. To date, the most reliable surveillance tool for amplification and timely indication of actual viral transmission available to mosquito control programs is a long-term sentinel chicken flock (Day, 1989). Mosquito control programs have also become highly effective in monitoring mosquito vector populations and dispersal. And with the advent of the internet, monitoring local environmental conditions has never been easier or more accurate. The one factor in epidemics that is not easily measured is the properties of the avian host populations.

The importance of the avian hosts in the amplification phase of SLEV and WNV in Florida has been demonstrated in various studies (McLean and Bowen 1980; Shaman and Day, 2002). Specifically, four species of birds have been identified to play the most importance in the amplification cycle: Blue Jay, Common Grackle, Mourning Dove, and Northern Cardinal (Day 2001, Day and Stark 1999). The distributions of these four important avian species are shown in **Figure 1**. Depending upon environmental conditions, these host species can produce large numbers of susceptible offspring, leading to increased arbovirus amplification and the possibility of reaching the primer threshold (Day, 2001).

To accurately gauge avian host amplification levels, host serosurveys are generally conducted. The primary downside to such serosurveys is that they are labor intensive. Other issues also include the requirement to obtain and keep the proper State and Federal permits, a commitment to establish baseline data sets though long-term sampling, and a general lack of experience in accurately interpreting the results (Day and Lewis, 1991). Due to such issues, host serosurveys are typically beyond the scope of most mosquito control surveillance programs in Florida.

While increased amplification is no guarantee of transmission to humans, any increased amplification of arbovirus in wild birds may increase the relative risk of transmission to humans and must be included in any serious risk analysis protocol. However, host amplification levels are only one of many wild bird factors that must be considered. Other factors, such as avian breeding success, percentage of the population that is susceptible, distribution, and relative abundance can be equally important to any risk analysis. The inability of mosquito control programs in Florida to monitor avian host factors leaves a large deficit in any effort by their arboviral surveillance programs to effectively evaluate risk to the human population.

The North American Breeding Bird Survey (BBS) provides data specific to population change and relative abundance for ~420 bird species in the United States and Canada (Sauer et al. 2008). Observers conduct fifty counts of 3 minutes in length, along predefined routes. Within the Peninsular Florida (PF) Bird Conservation Region (BCR) (**Figure 2**) the BBS consists of 75 routes with each route being ~39 km long with stops at approximately 0.8 km intervals. At each stop, every bird seen within a 0.4 km radius or heard is counted (BBS website, http://www.mbr-pwrc.usgs.gov/bbs/bbs.html). The BBS surveys are conducted during the peak of the nesting season, May – June) in Peninsular Florida. Data collected annually includes the total number of individuals (count) detected of the species and a modeled population index (trend). Those data are available online on the BBS website: https://www.pwrc.usgs.gov/BBS.

Starting in 1978, a state-wide sentinel chicken surveillance program was established in 36 Florida counties (Day and Lewis 1991) across Florida. Currently, there are over 40 organizations participating in the Florida sentinel chicken surveillance program, with approximately 340 sentinel chicken flocks located throughout the state. Current test protocols for sentinel chickens in Florida recommend that serum be drawn once a week from each chicken in each flock (Florida Department of Health, 2014). The collected serum samples are then shipped overnight to the Florida Department of Health (DOH) Bureau of Laboratory Services Activities (BLSA) in Tampa, FL for testing of the following viruses: eastern equine encephalitis virus (EEEV), Highlands J virus (HJV), St. Louis encephalitis virus (SLEV), and West Nile virus (WNV) via hemagglutination inhibition assay (HAI). Chickens with HAI titers of >1:10 are considered first time positive (FTP) test results.

The overall objective of this study was to evaluate the long-term association between and identify any temporal correlations between BBS data for the Blue Jay (BJ), Common Grackle (CG), Mourning Dove (MD), Northern Cardinal (NC), and Northern Mockingbird (NM) species and SLE virus transmission to sentinel chickens. Christmas Bird Count (CBC) data was also considered for analysis, but discarded based upon the timing of the counts being outside the main amplification and transmission season of SLE in Peninsular Florida. A time series analysis was used to explore the statistical relationship between the datasets at an annual frequency for the time period 1978 to 2015. Temporal correlations between the datasets were identified and analyzed using several statistical methods for the identified species and SLE seroconversions within the PF BCR. Identification of the statistical relationships, if any, between BBS data and SLE transmission in Florida may help with improvement of Arboviral transmission forecasting methods in support of mosquito control and statewide decision makers.

MATERIALS AND METHODS

Data Acquisition

Count data for Blue Jay, Common Grackle, Mourning Dove, Northern Cardinal, and Northern Mockingbird species were acquired for all 75 routes in the PF BCR via the BBS website. Spatially, raw counts are available at the individual route level and trend indexes are calculated at the BCR level. Aggregate count data and a BCR specific trend index were downloaded and prepared for analysis. The raw count data dataset covers the years 1966 to 2015 while the trend index consists of data from 1966 to 2013.

Weekly FTP seroconversions for SLE were acquired for the 37 counties located within the PF BCR. The dataset contains all FTP SLE seroconversions for the time period of 1978 to 2015 and was temporally aggregated to an annual level to match the BBS dataset.

Preliminary Data Analysis

In accordance with the study goals, an exploratory routine was used to determine the suitability of different Breeding Bird Survey datasets for time series analysis with the sentinel chicken seroconversion dataset. Breeding Bird Survey data raw counts and trend indexes were plotted and visually checked for changes in mean and variance (trend nonstationary) over time. The raw count datasets appeared to have stationary variance values but did exhibit a problematic change (increase) in mean values prior to 1987 and 1988 and after. Further research revealed that in 1988 the BBS saw an increase in data collection effort which caused the change in mean values between the two time periods. The modeled trend index data also appeared to have stationary variance values and did not exhibit the same change in means between the time periods that the raw count data exhibited. However, the trend index data did exhibit long term changes in mean values over the complete dataset time period. The long-term change was not unexpected as the modeled trend index values are used as an indicator of the abundance of bird species over time (trend). Due to the issues with both the raw counts and trend index datasets, a 3rd dataset was created for the analysis. Annual *averages per route* for the BCR were calculated using the raw count dataset. Visual exploration of plots of the calculated averages revealed more stationary mean and variance values than the raw count or trend index datasets. As such, the BBS calculated **averages** and modeled **trends** data sets were chosen for inclusion in the study along with the SLE seroconversions datasets. Spatially, BBS routes aligned poorly with a set of four counties that were initially chosen for the study (Indian River, Orange, Pinellas, and Volusia) (**Figure 2**). Due to the poor spatial alignment, it was decided to analyze the data at the PF BCR level to include a broader range of counties (37). A fifth bird species (Northern Mocking bird) was included in the analysis for broader scope, based upon the changes made to the study plan. The SLE seroconversions datasets were log-transformed before analysis to normalize the distribution and minimize standard error.

Time Series Analysis

The following datasets (1978 to 2015) were used in this analysis:

1. Blue Jay Average per Route - (1978 to 2015)
2. Common Grackle Average per Route - (1978 to 2015)
3. Mourning Dove Average per Route - (1978 to 2015)
4. Northern Cardinal Average per Route - (1978 to 2015)
5. Northern Mockingbird Average per Route - (1978 to 2015)
6. SLE Sentinel Chicken Seroconversions - (1978 to 2015)
7. Blue Jay Trend Index - (1978 to 2013)
8. Common Grackle Trend Index - (1978 to 2013)
9. Mourning Dove Trend Index - (1978 to 2013)
10. Northern Cardinal Trend Index - (1978 to 2013)
11. Northern Mockingbird Trend Index - (1978 to 2013)

The methods used in this research followed a modified ARIMA (Box-Jenkins) Modeling Algorithm for Time Series.

1. Check Time Series for Stable Variance and Normal Distribution
2. Transform Time Series if necessary
3. Check Time Series for Trends
4. Apply Regular and/or Seasonal Differencing if necessary
5. Select ARIMA Model for Independent Variable (*x*)
6. Check ARIMA Model Residuals for Correlation and Normal Distribution
7. Modify Model and/or Model Parameter Values if necessary
8. Fit Independent Variable (*x*) ARIMA Model to Dependent Variable (*y*) to prewhiten
9. Calculate Cross Correlation Function Values for all Time Series Lag Values
10. Calculate Granger Causality for any Significant Cross Correlation Function Values

The annual SLE Seroconversions Count was treated as dependent variable, and species specific BBS Averages and Trend Indexes were independent variables (10 total). Within the PF BCR, the statistical relationship between BBS count and SLE transmission over the period 1976 to 2015 and between BBS index and SLE transmission over the period 1976 to 2013 was examined. To account for auto-correlation among both dependent and independent variables over time, auto-regressive integrated moving average (ARIMA) models were fitted to each independent variable and use to pre-whiten all variables. Cross-correlations were then calculated up to ten time lags and examined for statistical significance. Significantly cross-correlated time lags were then tested for Granger causality to determine the ability of BBS data to predict changes in SLE transmission to sentinel chickens, or, in some cases, the ability of SLE transmission to sentinel chickens to predict changes in BBS data.

The detailed statistical methods of the analysis can be found at the end of this report in the section titled BBS Analysis.

RESULTS

For cross-correlation analysis with pre-whitening, ARIMA models were fitted to each variable as listed in **Table 1**. The purpose of pre-whitening is to eliminate any spurious correlations that may exist within or between datasets caused by either autocorrelation or similar trends within the data. The trend index datasets tended to exhibit better fits by the ARIMA models in general based upon the AIC scores. Each dataset was fitted for several ARIMA models and the AIC was used as the criteria for the best fit. Only the MD Index was fitted with an auto-regressive terms and all datasets exhibited a need for differencing via the Integrated term of the ARIMA models. All model residual auto-correlation and partial auto-correlation plots resembled white noise, indicating a god fit of the ARIMA models.

The cross-correlation coefficients for the association between pre-whitened BBS data and SLE transmission can be observed in **Figures 7 – 16**. All cross-correlation plots exhibited random fluctuations, indicating effective pre-whitening. The only plot with any significant correlations of interest was the CG avg x SLE plot (**Figure 8**), which indicated a significant positive correlation between SLE transmission to sentinel chickens and Common Grackle averages four and five years prior. The positive correlations were admittedly weak with coefficients just above 0.3 at a 95% confidence level. No other significant positive or negative correlations indicating a correlation between SLE transmission to sentinel chickens and prior BBS average or index data is visible within the plots. The plots do indicate various (9 total) significant correlations between BBS average or index data and prior SLE transmission to sentinel chickens, specifically Blue Jay average, Northern Cardinal average, Northern Mockingbird average, and Common Grackle index.

Both Granger causality testing results of SLE to previous BBS and BBS to previous SLE correlations are listed in **Table 2**. Only the Northern Mockingbird BBS to previous SLE nine years prior was significant.

DATASET	ARIMA MODEL	AIC
BLUE JAY AVERAGE	ARIMA(0,1,1)	206.89
COMMON GRACKLE AVERAGE	ARIMA(0,1,1)	286.71
MOURNING DOVE AVERAGE	ARIMA(0,1,1)	246.2
NORTHERN CARDINAL AVERAGE	ARIMA(0,1,0)	229.48
NORTHERN MOCKINGBIRD AVERAGE	ARIMA(0,1,1)	335.49
BLUE JAY INDEX	ARIMA(0,1,1)	71.27
COMMON GRACKLE INDEX	ARIMA(0,1,1)	190.43
MOURNING DOVE INDEX	ARIMA(1,1,0)	202.98
NORTHERN CARDINAL INDEX	ARIMA(0,1,2)	126.69
NORTHERN MOCKINGBIRD INDEX	ARIMA(0,1,0)	133.51

Table 1.

Correlation	Years Prior	Correlation Coefficient	Granger Causality Pr(>F)
SLE transmission leads Blue Jay average	8	+0.364*	0.356
Common Grackle average leads SLE transmission	4	+0.325*	0.4088
Common Grackle average leads SLE transmission	5	+0.353*	0. 4727
SLE transmission leads Common Grackle average	9	0.347*	0.1595
SLE transmission leads Northern Cardinal average	2	-0.426*	0. 2672
SLE transmission leads Northern Cardinal average	8	0.323*	0. 4819
SLE transmission leads Northern Cardinal average	9	-0.419*	0. 363
SLE transmission leads Northern Mockingbird average	8	-0.363*	0. 3234
SLE transmission leads Northern Mockingbird average	9	0.336*	0. 0352*
SLE transmission leads Common Grackle index	2	0.467*	0.1595

* significant at 0.05 level

Table 2.

DISCUSSION

The relationship between avian hosts and SLE transmission in Florida is a complex one with the effect(s) of year to year fluctuations of avian host populations on the transmission of SLE not well documented. While avian bird populations play a pivotal role in the SLE amplification and transmission cycle in Florida, quantifying this role can be challenging. Research has shown that environmental factors are the driving force behind high SLE transmission years as they drive avian and mosquito populations together for increased amplification. This study hoped to reveal quantifiable correlations between BBS data and SLE seroconversions across Peninsular Florida for two reasons: (1) Identifying associations between bird data and SLE transmission may be helpful in forecasting and prevention of human SLE cases and (2) determination of the direction, strength, and time differences of the associations may be useful in furthering our understanding of the SLE amplification/transmission cycle.

The results of the study showed a significant positive correlation between BBS Common Grackles annual averages and SLE annual counts of transmission to sentinel chickens across Peninsular Florida 4 and 5 years prior. This result indicates that increasing Common Grackles abundance is associated with increased SLE transmission to sentinel chickens 4 to 5 years later. A biological explanation for the association is not readily apparent but one explanation would be a very small statistical co-integration of the datasets due to similar driving forces such as environmental conditions.

The apparent lack of significant positive correlations between SLE transmission to sentinel chickens and prior BBS data counts for four of the five species examined is not surprising. Identifying correlations in time series datasets can be difficult when the sample size is below 50 and determining causality of any identified correlations can be tenuous at best. Statistically, checking for correlations between time series datasets can be problematic. Feedback loops between datasets and autocorrelations within datasets must be eliminated through the process of noise whitening. While most testing methods for autocorrelations are considered robust for larger (>50) datasets, some statisticians have argued that autocorrelations in smaller datasets, even after pre-whitening, can be hard to detect.

Our analysis indicated decreased SLE transmission to sentinel chickens lead decreases in Blue Jay abundance at 8 years, Common Grackle abundance at 2 and 9 years, Northern Cardinal abundance at 2, 8, and 9 years, and Northern Mockingbird abundance at 8 and 9 years. The only correlation exhibiting any Granger causality was the Northern Mockingbird at 9 years. Some research has implied that West Nile virus has had an ecological impact on various North American bird populations (LaDeau et al. -2007, Koenig et al. – 2007) . Could this explain the significant negative correlations of SLE transmission leading BBS avian abundance averages and trend indexes? It would be hard to imagine any significant impact upon wild bird populations by SLE transmission levels at 8 or 9 years. The associations at 2 years would be more likely, but one would expect to see a range of years (1 to 3) that would have correlations. Again, biological explanations for these associations are not readily apparent.

The purpose of the study was to provide evidence of any strong signals between BBS count data and SLE transmission to sentinel chickens. With a lack of any strong associations between the datasets we must assume one of two ideas. First, if we assume bird abundance does have a strong association or effect on SLE transmission to sentinel chickens, then the BBS data may not be well suited to capture the nuances of bird abundance that influence or drive SLE transmission. Second, if we assume that assume that bird abundance does NOT have a strong association or effect on SLE transmission to sentinel chickens, then perhaps the bird abundance is only part of the equation. Again, the factors that drive amplification and transmission of SLE in Peninsular Florida are integrated in a complex manner in which the strength on any one factor may not lead to an increase in SLE transmission. The abundance of specific arboviruses, and mosquito or bird species in any given year will not drive SLE amplification and transmission. Only when they are aligned spatially and temporally by driving environmental conditions do we see an increase in the transmission of SLE. An approach to deriving these complex integrations between virus, mosquito, avian, and environmental factors is the collection of more data over time. There can be no doubt that this study highlights the importance of the sentinel chicken programs throughout Florida is giving us important data to combat the spread of arboviruses.

ACKNOWLEDGMENTS

This project was funded by the Florida Legislature. We would like to acknowledge the following organizations and people for these help with this project: Florida Department of Agriculture and Consumer Services, Florida Department of Health, Timothy Hope, and Daniel Forsythe.

REFERENCES

Day, J. F. 1989. The use of sentinel chickens for arbovirus surveillance in Florida. J.Fl.Anti-Mosq.Assoc., 60(2), 56–61.

Day, J.F. 1991. A review of the 1990 St. Louis encephalitis virus epidemic in Indian River County, Florida. Proceedings of the New Jersey Mosquito Control Association 78:32-39.

Day, J. F., & Lewis, A. L. 1991. An integrated approach to arboviral surveillance in Indian River County, Florida. Journal of the Florida Mosquito Control Association, 62(2)(2), 46–52.

Day, J.F. 2001. Predicting St. Louis encephalitis virus epidemics: Lessons from recent, and not so recent, outbreaks. Annual Review of Entomol. 46:111-38.

Day, J.F. and J. Shaman. 2008. Using hydrologic conditions to track the risk of focal and epidemic arboviral transmission in peninsular Florida. J. Med. Entomol. 45:458-469.

Day, J.F. and L.M. Stark. 1999. Avian serology in a St. Louis encephalitis epicenter before during, and after a widespread epidemic in South Florida, USA. J. Med. Entomol. 36:626-633.

Day, J. F., Ross, G., & Connelly, R. C. 2013. TECHNICAL BULLETIN OF THE FLORIDA MOSQUITO CONTROL ASSOCIATION, 9.

E., A.-Y. (2015). Application of ARIMA Models in Forecasting Monthly Average Surface Temperature of Brong Ahafo Region of Ghana. International Journal of Statistics and Applications, 5(5), 237–246. http://doi.org/10.5923/j.statistics.20150505.08

Florida Department of Health, Surveillance and Control of Selected Mosquito-borne Diseases in Florida, 2014

Hyndman, R. J., & Khandakar, Y. (2008). Automatic time series forecasting: The forecast package for R. Journal Of Statistical Software, 27(3), C3–C3. http://doi.org/10.18637/jss.v027.i03

Koenig, W. D., L. Marcus, T. W. Scott, and J. L. Dickinson. 2007. West Nile virus and California breeding bird declines. EcoHealth 4(1):18-24

LaDeau, Shannon L., A. Marm Kilpatrick, and Peter P. Mara. 2007. West Nile virus emergence and large-scale declines of North American bird populations. Nature (London) 447(7145):710-713.

Sauer, J. R., J. E. Fallon, and R. Johnson. 2003. Use of North American Breeding Bird Survey data to estimate population change for bird conservation regions. Journal of Wildlife Management 67(2):372-389.

Sauer, J.R. and W. A. Link. 2002. Hierarchical Modeling of Population Stability and Species Group Attributes from Survey Data. Ecology,Vol. 83, No. 6 (Jun., 2002), pp. 1743-1751

Sauer, J. R., W. A. Link, W. L. Kendall, J. R. Kelley, and D. K. Niven. 2008. A hierarchical model for estimating change in American Woodcock populations. Journal of Wildlife Management 72(1):204-214.

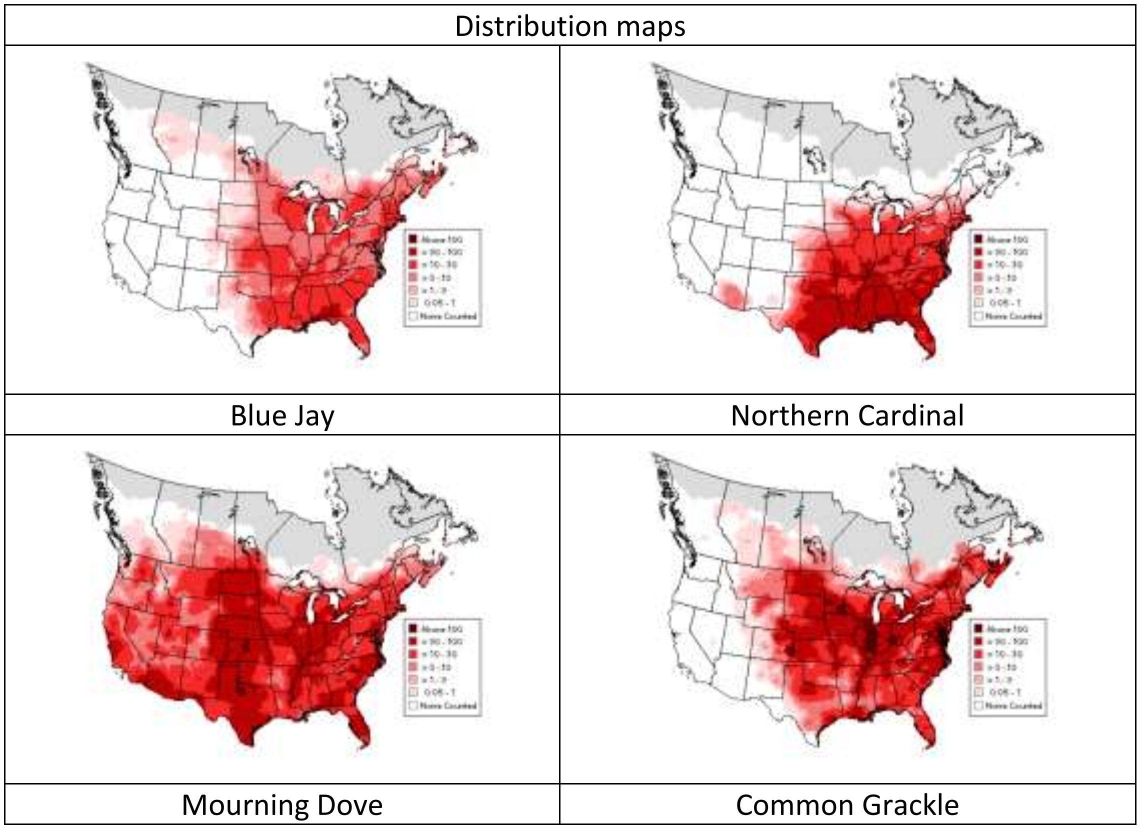
Shaman, J., J.F. Day, and M. Stieglitz. 2002. Drought-Induced Amplification of Saint Louis encephalitis virus, Florida. Emerging Infectious Diseases. Vol. 8, No. 6. pp. 575-580.

Shaman, J., J.F. Day, and M. Stieglitz. 2003. St. Louis encephalitis virus in wild birds during the 1990 south Florida epidemic: the importance of drought, wetting conditions, and the emergence of Culex nigripalpus to arboviral amplification and transmission. J. Med. Entomol. 40:547-554.

Zucchini, W. (2011). Time Series Analysis with R. Handbook of Statistics, 30(February), 1–23. http://doi.org/http://dx.doi.org/10.1016/B978-0-444-53858-1.00023-5

FIGURES

Figure 1.



Maps produced by the USGS Patuxent Wildlife Research Center at http://www.mbr-pwrc.usgs.gov/bbs/

Figure 2.

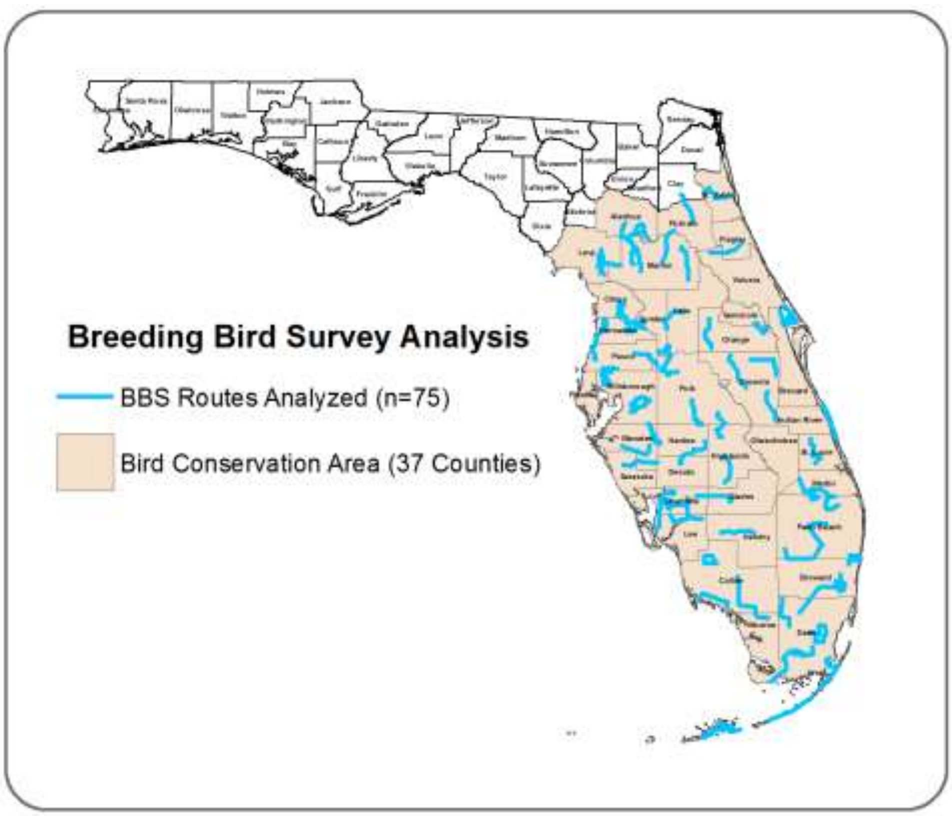


Figure 3.

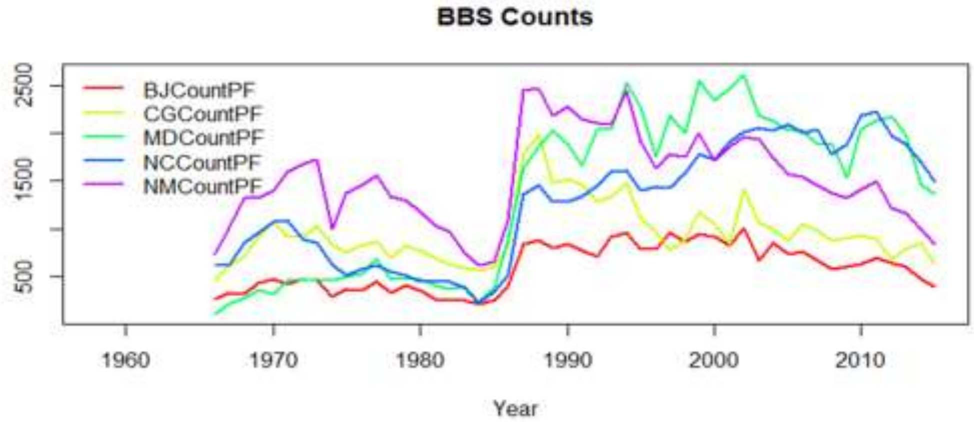


Figure 4.

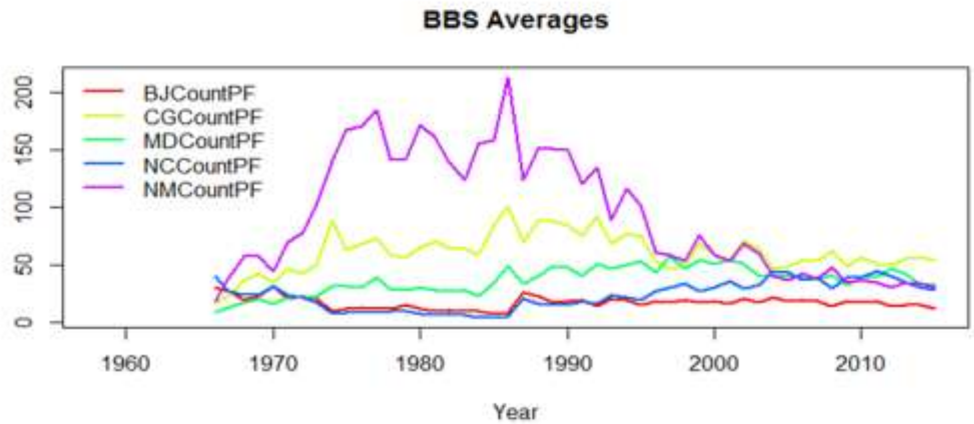


Figure 5.

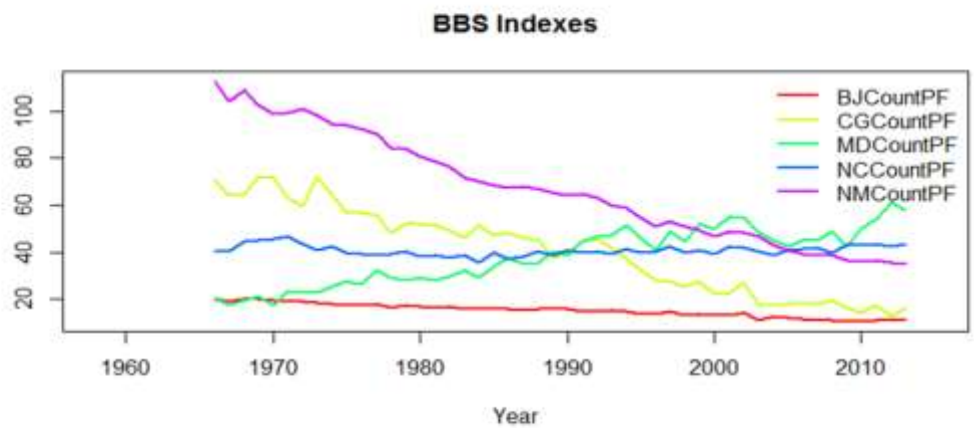
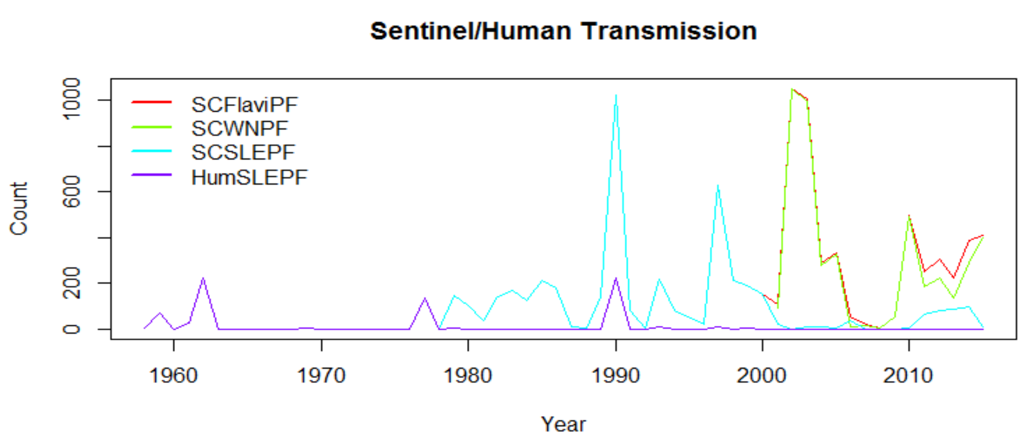


Figure 6.



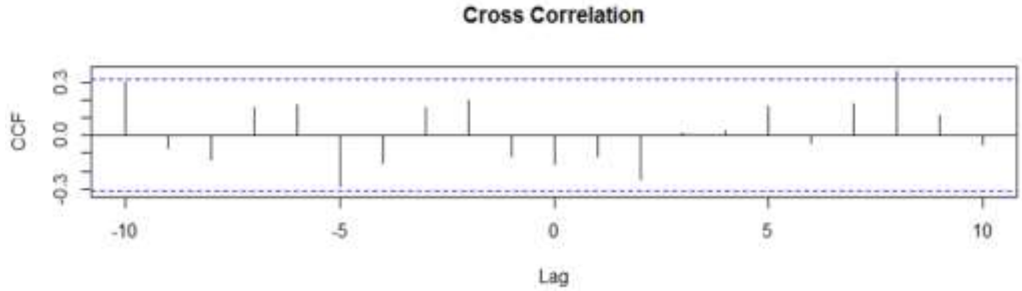


Figure 7.
BJ avg x SLE

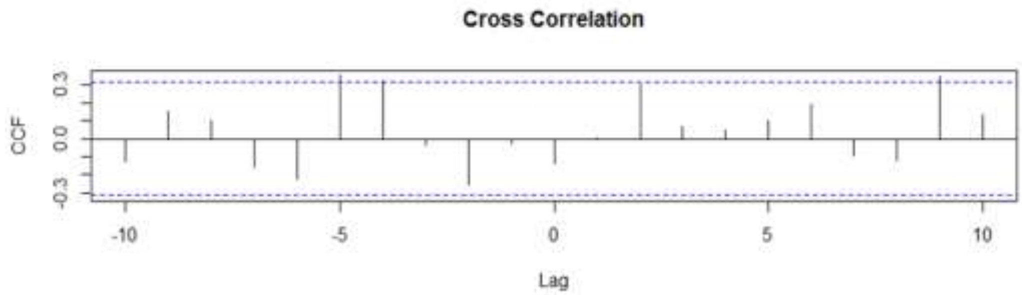


Figure 8.
CG avg x SLE

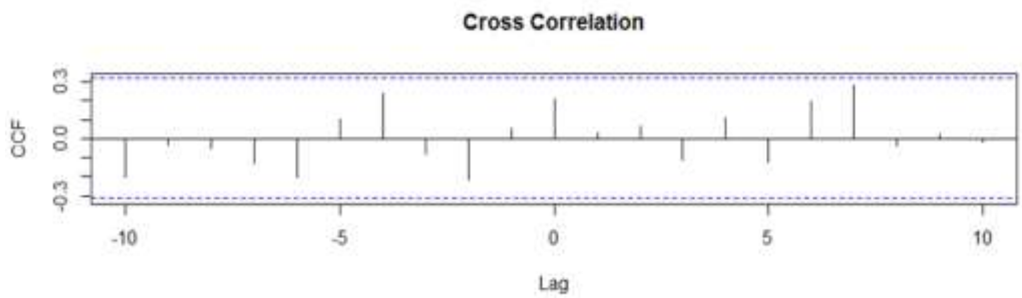


Figure 9.
MD avg x SLE

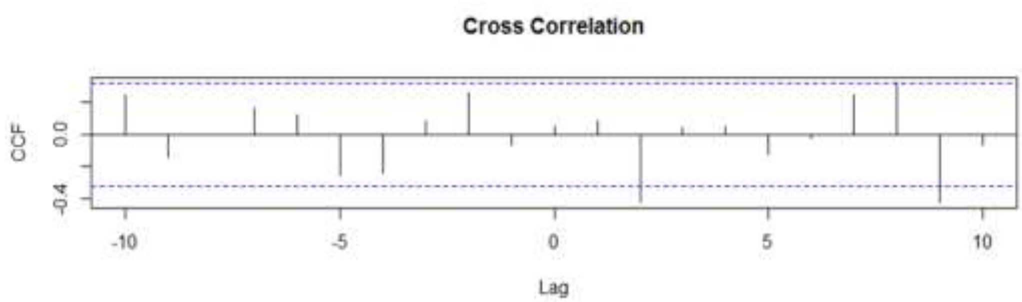


Figure 10.
NC avg x SLE

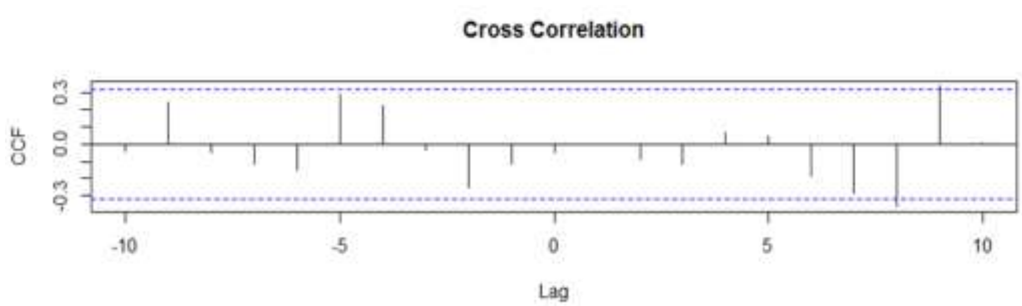


Figure 11.
NM avg x SLE

Figure 12.
BJ ind x SLE

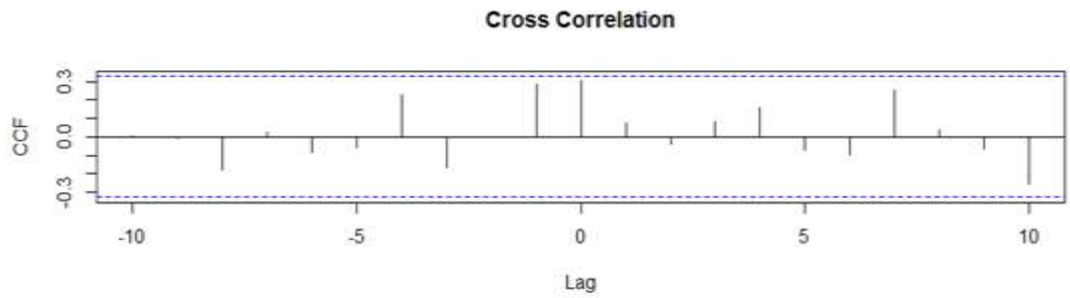


Figure 13.
CG ind x SLE

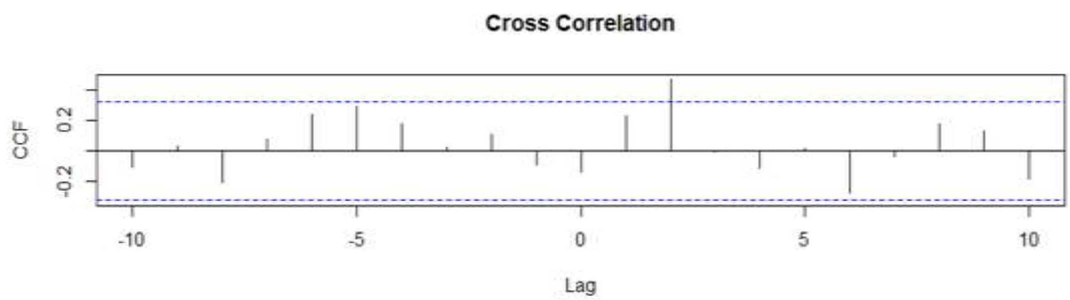


Figure 14.
MD ind x SLE

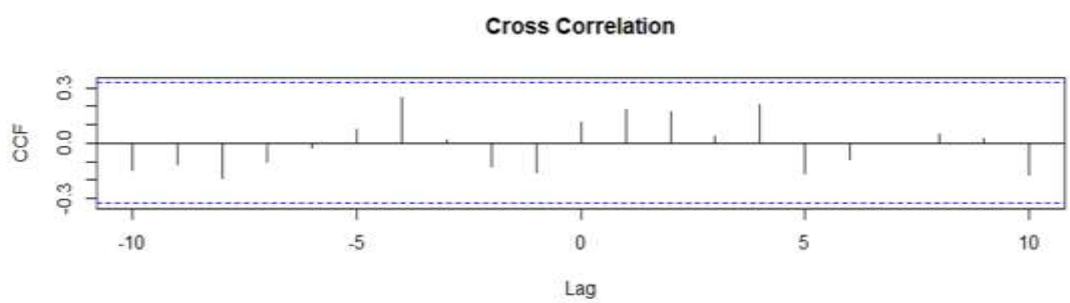


Figure 15.
NC ind x SLE

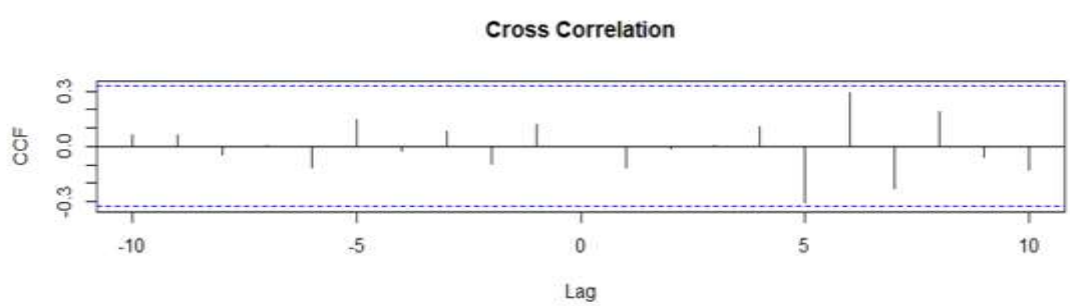
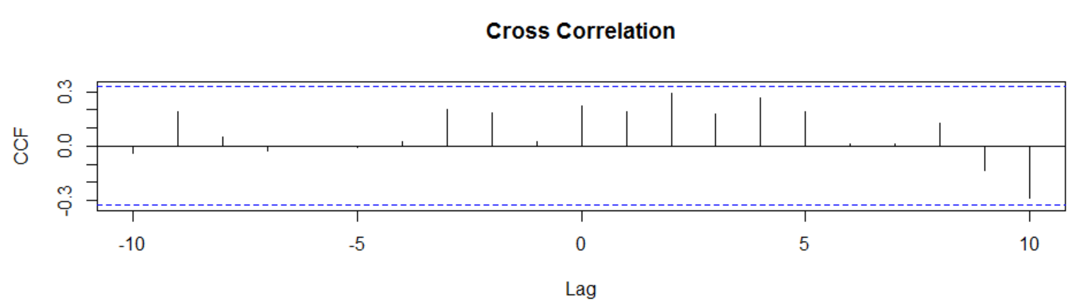


Figure 16.
NM ind x LE



BBS Analysis

Jonathan F Day

Gregory K Ross

August 8, 2016

This study is done in a reproducible research manner. All statistical analysis and figures below were created in the open source software package R.

Setup

Load packages, get dataset, and load variables. Dataset consists of a csv file with annual counts of data. Dataste is then filtered and subsetted into various time series. Variables are then created from the time series. All Variables are Annual Counts.

```
knitr::opts_chunk$set(echo = TRUE, warning=FALSE, message=FALSE, comment = NA)
options(width = 90)

# list of needed packages
list.of.packages <- c("tsoutliers", "nortest", "fitdistrplus", "tseries", "forecast",
                      "TSA", "egcm", "lmtest", "gplots", "forecast", "moments", "RCurl", "pander")

# list of packages that are not installed
new.packages <- list.of.packages[!(list.of.packages %in% installed.packages()
  [,"Package"])]

# install needed packages
if(length(new.packages)) install.packages(new.packages,
  repos = "http://cran.us.r-project.org")

library("tsoutliers", lib.loc="~/R/win-library/3.2")
library("forecast", lib.loc="~/R/win-library/3.2")
library("nortest", lib.loc="~/R/win-library/3.2")
library("fitdistrplus", lib.loc="~/R/win-library/3.2")
library("tseries", lib.loc="~/R/win-library/3.2")
library("TSA", lib.loc="~/R/win-library/3.2")
library("egcm", lib.loc="~/R/win-library/3.2")
library("lmtest", lib.loc="~/R/win-library/3.2")
library("gplots", lib.loc="~/R/win-library/3.2")
library("stats", lib.loc="C:/Program Files/Microsoft/MRO/R-3.2.3/library")
library("moments", lib.loc="~/R/win-library/3.2")
library("RCurl", lib.loc="~/R/win-library/3.2")
library("pander", lib.loc="~/R/win-library/3.2")
# List information about R software session
sessionInfo()
```

```
## R version 3.2.3 (2015-12-10)
## Platform: i386-w64-mingw32/i386 (32-bit)
## Running under: Windows 10 x64 (build 10240)
##
## locale:
## [1] LC_COLLATE=English_United States.1252  LC_CTYPE=English_United States.1252
## [3] LC_MONETARY=English_United States.1252 LC_NUMERIC=C
## [5] LC_TIME=English_United States.1252
##
## attached base packages:
## [1] stats     graphics  grDevices utils     datasets  methods   base
##
## other attached packages:
##  [1] pander_0.6.0      RCurl_1.95-4.8     bitops_1.0-6      moments_0.14
##  [5] gplots_2.17.0     lmtest_0.9-34      egcm_1.0.8        TTR_0.23-0
```

```
##  [9] xts_0.9-7          TSA_1.01           mgcv_1.8-9         nlme_3.1-122
## [13] locfit_1.5-9.1     leaps_2.9          tseries_0.10-34    fitdistrplus_1.0-6
## [17] MASS_7.3-45        nortest_1.0-4      forecast_6.2       timeDate_3012.100
## [21] zoo_1.7-12         tsoutliers_0.6
##
## loaded via a namespace (and not attached):
##  [1] gdata_2.17.0          ggplot2_2.0.0         timeSeries_3022.101.2
##  [4] lattice_0.20-33       formatR_1.2.1         stringr_1.0.0
##  [7] gtools_3.5.0          parallel_3.2.3        polynom_1.3-8
## [10] Rcpp_0.12.6           plyr_1.8.3            tools_3.2.3
## [13] rmarkdown_1.0.9002    fArma_3010.79         knitr_1.13
## [16] scales_0.3.0          nnet_7.3-11           digest_0.6.8
## [19] evaluate_0.9          gtable_0.1.2          stsm_1.7
## [22] KernSmooth_2.23-15    Matrix_1.2-3          stringi_1.1.1
## [25] caTools_1.17.1        fBasics_3011.87       htmltools_0.3.5
## [28] munsell_0.4.2         grid_3.2.3            colorspace_1.2-6
## [31] fracdiff_1.4-2        quadprog_1.5-5        magrittr_1.5
## [34] splines_3.2.3         yaml_2.1.13           survival_2.38-3
## [37] urca_1.2-8            KFKSDS_1.6
```

```r
# clear working environment
rm(list = ls())

# Load data from online respository
bbs = read.csv("https://dl.dropboxusercontent.com/u/10866552/research/BBS/bbs.csv")
attach(bbs)

# create filtered dataframes
bbs66to13<- subset(bbs, bbs$Year>1965 & bbs$Year<2014)
bbs66to15<- subset(bbs, bbs$Year>1965)
bbs78to13<- subset(bbs, bbs$Year>1977 & bbs$Year<2014)
bbs78to15<- subset(bbs, bbs$Year>1977)
bbs01to13<- subset(bbs, bbs$Year>2000 & bbs$Year<2014)
bbs01to15<- subset(bbs, bbs$Year>2000)

# create filtered time series

# 1958 to 2015 Times series
bbs58to15ts <- ts(bbs, start = 1958, end = 2015, frequency = 1)
# 1966 to 2013 Times series
bbs66to13ts <- ts(bbs66to13, start = 1966, end = 2013, frequency = 1)
# 1966 to 2015 Times series
bbs66to15ts <- ts(bbs66to15, start = 1966, end = 2015, frequency = 1)
# 1978 to 2013 Times series
bbs78to13ts <- ts(bbs78to13, start = 1978, end = 2013, frequency = 1)
# 1978 to 2015 Times series
bbs78to15ts <- ts(bbs78to15, start = 1978, end = 2015, frequency = 1)
# 2001 to 2013 Times series
bbs01to13ts <- ts(bbs78to13, start = 2001, end = 2013, frequency = 1)
# 2001 to 2015 Times series
bbs01to15ts <- ts(bbs78to15, start = 2001, end = 2015, frequency = 1)

# BBS Variables
# 1966 to 2015 Counts
# Blue Jay Counts 1966 to 2015
BJCountPF66to15<- subset(bbs$BJCountPF, bbs$Year>1965)
# Blue Jay Counts 1966 to 2015 Time Series
BJCountPF66to15ts<- ts(BJCountPF66to15, start = 1966, end = 2015, frequency = 1)
# Common Grackle Counts 1966 to 2015
CGCountPF66to15<- subset(bbs$CGCountPF, bbs$Year>1965)
# Common Grackle Counts 1966 to 2015 Time Series
CGCountPF66to15ts<- ts(CGCountPF66to15, start = 1966, end = 2015, frequency = 1)
# Mourning Dove Counts 1966 to 2015
MDCountPF66to15<- subset(bbs$MDCountPF, bbs$Year>1965)
# Mourning Dove Counts 1966 to 2015 Time Series
```

```
MDCountPF66to15ts<- ts(MDCountPF66to15, start = 1966, end = 2015, frequency = 1)
# Northern Cardinal Counts 1966 to 2015
NCCountPF66to15<- subset(bbs$NCCountPF, bbs$Year>1965)
# Northern Cardinal Counts 1966 to 2015 Time Series
NCCountPF66to15ts<- ts(NCCountPF66to15, start = 1966, end = 2015, frequency = 1)
# Northern Mockingbird Counts 1966 to 2015
NMCountPF66to15<- subset(bbs$NMCountPF, bbs$Year>1965)
# Northern Mockingbird Counts 1966 to 2015 Time Series
NMCountPF66to15ts<- ts(NMCountPF66to15, start = 1966, end = 2015, frequency = 1)

# 1978 to 2015 Counts
# Blue Jay Counts 1978 to 2015
BJCountPF78to15<- subset(bbs$BJCountPF, bbs$Year>1977)
# Blue Jay Counts 1978 to 2015 Time Series
BJCountPF78to15ts<- ts(BJCountPF78to15, start = 1978, end = 2015, frequency = 1)
# Common Grackle Counts 1978 to 2015 Time Series
CGCountPF78to15<- subset(bbs$CGCountPF, bbs$Year>1977)
# Common Grackle Counts 1978 to 2015 Time Series
CGCountPF78to15ts<- ts(CGCountPF78to15, start = 1978, end = 2015, frequency = 1)
# Mourning Dove Counts 1978 to 2015
MDCountPF78to15<- subset(bbs$MDCountPF, bbs$Year>1977)
# Mourning Dove Counts 1978 to 2015 Time Series
MDCountPF78to15ts<- ts(MDCountPF78to15, start = 1978, end = 2015, frequency = 1)
# Northern Cardinal Counts 1978 to 2015
NCCountPF78to15<- subset(bbs$NCCountPF, bbs$Year>1977)
# Northern Cardinal Counts 1978 to 2015 Time Series
NCCountPF78to15ts<- ts(NCCountPF78to15, start = 1978, end = 2015, frequency = 1)
# Northern Mockingbird Counts 1978 to 2015
NMCountPF78to15<- subset(bbs$NMCountPF, bbs$Year>1977)
# Northern Mockingbird Counts 1978 to 2015 Time Series
NMCountPF78to15ts<- ts(NMCountPF78to15, start = 1978, end = 2015, frequency = 1)

# 2001 to 2015 Counts
# Blue Jay Counts 2001 to 2015
BJCountPF01to15<- subset(bbs$BJCountPF, bbs$Year>2000)
# Blue Jay Counts 2001 to 2015 Time Series
BJCountPF01to15ts<- ts(BJCountPF01to15, start = 2001, end = 2015, frequency = 1)
# Common Grackle Counts 2001 to 2015
CGCountPF01to15<- subset(bbs$CGCountPF, bbs$Year>2000)
# Common Grackle Counts 2001 to 2015 Time Series
CGCountPF01to15ts<- ts(CGCountPF01to15, start = 2001, end = 2015, frequency = 1)
# Mourning Dove Counts 2001 to 2015
MDCountPF01to15<- subset(bbs$MDCountPF, bbs$Year>2000)
# Mourning Dove Counts 2001 to 2015 Time Series
MDCountPF01to15ts<- ts(MDCountPF01to15, start = 2001, end = 2015, frequency = 1)
# Northern Cardinal Counts 2001 to 2015
NCCountPF01to15<- subset(bbs$NCCountPF, bbs$Year>2000)
# Northern Cardinal Counts 2001 to 2015 Time Series
NCCountPF01to15ts<- ts(NCCountPF01to15, start = 2001, end = 2015, frequency = 1)
# Northern Mockingbird Counts 2001 to 2015
NMCountPF01to15<- subset(bbs$NMCountPF, bbs$Year>2000)
# Northern Mockingbird Counts 2001 to 2015 Time Series
NMCountPF01to15ts<- ts(NMCountPF01to15, start = 2001, end = 2015, frequency = 1)

# 1966 to 2015 Averages
# Blue Jay Averages 1966 to 2015
BJAveragePF66to15<- subset(bbs$BJAveragePF, bbs$Year>1965)
# Blue Jay Averages 1966 to 2015 Time Series
BJAveragePF66to15ts<- ts(BJAveragePF66to15, start = 1966, end = 2015, frequency = 1)
# Common Grackle Averages 1966 to 2015
CGAveragePF66to15<- subset(bbs$CGAveragePF, bbs$Year>1965)
# Common Grackle Averages 1966 to 2015 Time Series
CGAveragePF66to15ts<- ts(CGAveragePF66to15, start = 1966, end = 2015, frequency = 1)
# Mourning Dove Averages 1966 to 2015
MDAveragePF66to15<- subset(bbs$MDAveragePF, bbs$Year>1965)
```

```r
# Mourning Dove Averages 1966 to 2015 Time Series
MDAveragePF66to15ts<- ts(MDAveragePF66to15, start = 1966, end = 2015, frequency = 1)
# Northern Cardinal Averages 1966 to 2015
NCAveragePF66to15<- subset(bbs$NCAveragePF, bbs$Year>1965)
# Northern Cardinal Averages 1966 to 2015 Time Series
NCAveragePF66to15ts<- ts(NCAveragePF66to15, start = 1966, end = 2015, frequency = 1)
# Northern Mockingbird Averages 1966 to 2015
NMAveragePF66to15<- subset(bbs$NMAveragePF, bbs$Year>1965)
# Northern Mockingbird Averages 1966 to 2015 Time Series
NMAveragePF66to15ts<- ts(NMAveragePF66to15, start = 1966, end = 2015, frequency = 1)

# 1978 to 2015 Averages
# Blue Jay Averages 1978 to 2015
BJAveragePF78to15<- subset(bbs$BJAveragePF, bbs$Year>1977)
# Blue Jay Averages 1978 to 2015 Time Series
BJAveragePF78to15ts<- ts(BJAveragePF78to15, start = 1978, end = 2015, frequency = 1)
# Common Grackle Averages 1978 to 2015
CGAveragePF78to15<- subset(bbs$CGAveragePF, bbs$Year>1977)
# Common Grackle Averages 1978 to 2015 Time Series
CGAveragePF78to15ts<- ts(CGAveragePF78to15, start = 1978, end = 2015, frequency = 1)
# Mourning Dove Averages 1978 to 2015
MDAveragePF78to15<- subset(bbs$MDAveragePF, bbs$Year>1977)
# Mourning Dove Averages 1978 to 2015 Time Series
MDAveragePF78to15ts<- ts(MDAveragePF78to15, start = 1978, end = 2015, frequency = 1)
# Northern Cardinal Averages 1978 to 2015
NCAveragePF78to15<- subset(bbs$NCAveragePF, bbs$Year>1977)
# Northern Cardinal Averages 1978 to 2015 Time Series
NCAveragePF78to15ts<- ts(NCAveragePF78to15, start = 1978, end = 2015, frequency = 1)
# Northern Mockingbird Averages 1978 to 2015
NMAveragePF78to15<- subset(bbs$NMAveragePF, bbs$Year>1977)
# Northern Mockingbird Averages 1978 to 2015 Time Series
NMAveragePF78to15ts<- ts(NMAveragePF78to15, start = 1978, end = 2015, frequency = 1)

# 2001 to 2015 Averages
# Blue Jay Averages 2001 to 2015
BJAveragePF01to15<- subset(bbs$BJAveragePF, bbs$Year>2000)
# Blue Jay Averages 2001 to 2015 Time Series
BJAveragePF01to15ts<- ts(BJAveragePF01to15, start = 2001, end = 2015, frequency = 1)
# Common Grackle Averages 2001 to 2015
CGAveragePF01to15<- subset(bbs$CGAveragePF, bbs$Year>2000)
# Common Grackle Averages 2001 to 2015 Time Series
CGAveragePF01to15ts<- ts(CGAveragePF01to15, start = 2001, end = 2015, frequency = 1)
# Mourning Dove Averages 2001 to 2015
MDAveragePF01to15<- subset(bbs$MDAveragePF, bbs$Year>2000)
# Mourning Dove Averages 2001 to 2015 Time Series
MDAveragePF01to15ts<- ts(MDAveragePF01to15, start = 2001, end = 2015, frequency = 1)
# Northern Cardinal Averages 2001 to 2015
NCAveragePF01to15<- subset(bbs$NCAveragePF, bbs$Year>2000)
# Northern Cardinal Averages 2001 to 2015 Time Series
NCAveragePF01to15ts<- ts(NCAveragePF01to15, start = 2001, end = 2015, frequency = 1)
# Northern Mockingbird Averages 2001 to 2015
NMAveragePF01to15<- subset(bbs$NMAveragePF, bbs$Year>2000)
# Northern Mockingbird Averages 2001 to 2015 Time Series
NMAveragePF01to15ts<- ts(NMAveragePF01to15, start = 2001, end = 2015, frequency = 1)

# 1966 to 2013 Index
# Blue Jay Index 1966 to 2013
BJIndexPF66to13<- subset(bbs$BJIndexPF, bbs$Year>1965 & bbs$Year<2014)
# Blue Jay Index 1966 to 2013 Time Series
BJIndexPF66to13ts<- ts(BJIndexPF66to13, start = 1966, end = 2013, frequency = 1)
# Common Grackle Index 1966 to 2013
CGIndexPF66to13<- subset(bbs$CGIndexPF, bbs$Year>1965 & bbs$Year<2014)
# Common Grackle Index 1966 to 2013 Time Series
CGIndexPF66to13ts<- ts(CGIndexPF66to13, start = 1966, end = 2013, frequency = 1)
# Mourning Dove Index 1966 to 2013
```

```
MDIndexPF66to13<- subset(bbs$MDIndexPF, bbs$Year>1965 & bbs$Year<2014)
# Mourning Dove Index 1966 to 2013 Time Series
MDIndexPF66to13ts<- ts(MDIndexPF66to13, start = 1966, end = 2013, frequency = 1)
# Northern Cardinal Index 1966 to 2013
NCIndexPF66to13<- subset(bbs$NCIndexPF, bbs$Year>1965 & bbs$Year<2014)
# Northern Cardinal Index 1966 to 2013 Time Series
NCIndexPF66to13ts<- ts(NCIndexPF66to13, start = 1966, end = 2013, frequency = 1)
# Northern Mockingbird Index 1966 to 2013
NMIndexPF66to13<- subset(bbs$NMIndexPF, bbs$Year>1965 & bbs$Year<2014)
# Northern Mockingbird Index 1966 to 2013 Time Series
NMIndexPF66to13ts<- ts(NMIndexPF66to13, start = 1966, end = 2013, frequency = 1)

# 1978 to 2013 Index
# Blue Jay Index 1978 to 2013
BJIndexPF78to13<- subset(bbs$BJIndexPF, bbs$Year>1977 & bbs$Year<2014)
# Blue Jay Index 1978 to 2013 Time Series
BJIndexPF78to13ts<- ts(BJIndexPF78to13, start = 1978, end = 2013, frequency = 1)
# Common Grackle Index 1978 to 2013 Time Series
CGIndexPF78to13<- subset(bbs$CGIndexPF, bbs$Year>1977 & bbs$Year<2014)
# Common Grackle Index 1978 to 2013 Time Series
CGIndexPF78to13ts<- ts(CGIndexPF78to13, start = 1978, end = 2013, frequency = 1)
# Mourning Dove Index 1978 to 2013
MDIndexPF78to13<- subset(bbs$MDIndexPF, bbs$Year>1977 & bbs$Year<2014)
# Mourning Dove Index 1978 to 2013 Time Series
MDIndexPF78to13ts<- ts(MDIndexPF78to13, start = 1978, end = 2013, frequency = 1)
# Northern Cardinal Index 1978 to 2013
NCIndexPF78to13<- subset(bbs$NCIndexPF, bbs$Year>1977 & bbs$Year<2014)
# Northern Cardinal Index 1978 to 2013 Time Series
NCIndexPF78to13ts<- ts(NCIndexPF78to13, start = 1978, end = 2013, frequency = 1)
# Northern Mockingbird Index 1978 to 2013
NMIndexPF78to13<- subset(bbs$NMIndexPF, bbs$Year>1977 & bbs$Year<2014)
# Northern Mockingbird Index 1978 to 2013 Time Series
NMIndexPF78to13ts<- ts(NMIndexPF78to13, start = 1978, end = 2013, frequency = 1)

# 2001 to 2013 Index
# Blue Jay Index 2001 to 2013
BJIndexPF01to13<- subset(bbs$BJIndexPF, bbs$Year>2000 & bbs$Year<2014)
# Blue Jay Index 2001 to 2013 Time Series
BJIndexPF01to13ts<- ts(BJIndexPF01to13, start = 2001, end = 2013, frequency = 1)
# Common Grackle Index 2001 to 2013
CGIndexPF01to13<- subset(bbs$CGIndexPF, bbs$Year>2000 & bbs$Year<2014)
# Common Grackle Index 2001 to 2013 Time Series
CGIndexPF01to13ts<- ts(CGIndexPF01to13, start = 2001, end = 2013, frequency = 1)
# Mourning Dove Index 2001 to 2013
MDIndexPF01to13<- subset(bbs$MDIndexPF, bbs$Year>2000 & bbs$Year<2014)
# Mourning Dove Index 2001 to 2013 Time Series
MDIndexPF01to13ts<- ts(MDIndexPF01to13, start = 2001, end = 2013, frequency = 1)
# Northern Cardinal Index 2001 to 2013
NCIndexPF01to13<- subset(bbs$NCIndexPF, bbs$Year>2000 & bbs$Year<2014)
# Northern Cardinal Index 2001 to 2013 Time Series
NCIndexPF01to13ts<- ts(NCIndexPF01to13, start = 2001, end = 2013, frequency = 1)
# Northern Mockingbird Index 2001 to 2013
NMIndexPF01to13<- subset(bbs$NMIndexPF, bbs$Year>2000 & bbs$Year<2014)
# Northern Mockingbird Indexs 2001 to 2013 Time Series
NMIndexPF01to13ts<- ts(NMIndexPF01to13, start = 2001, end = 2013, frequency = 1)

# Transmission variables
# Sentinel Chicken SLE/WN/Flavi Variables
# Sentinel Chicken SLE Seroconversion 1978 to 2015
SCSLEPF78to15<- subset(bbs$SCSLEPF, bbs$Year>1977)
# Sentinel Chicken SLE Seroconversion 1978 to 2015 Time Series
SCSLEPF78to15ts<- ts(SCSLEPF78to15, start = 1978, end = 2015, frequency = 1)
# Sentinel Chicken SLE Seroconversion 1978 to 2013
SCSLEPF78to13<- subset(bbs$SCSLEPF, bbs$Year>1977 & bbs$Year<2014)
# Sentinel Chicken SLE Seroconversion 1978 to 2013 Time Series
```

```r
SCSLEPF78to13ts<- ts(SCSLEPF78to13, start = 1978, end = 2013, frequency = 1)
# Sentinel Chicken SLE Seroconversion 2001 to 2015
SCSLEPF01to15<- subset(bbs$SCSLEPF, bbs$Year>2000)
# Sentinel Chicken SLE Seroconversion 2001 to 2015 Time Series
SCSLEPF01to15ts<- ts(SCSLEPF01to15, start = 2001, end = 2015, frequency = 1)
# Sentinel Chicken WN Seroconversion 1978 to 2015
SCWNPF78to15<- subset(bbs$SCWNPF, bbs$Year>1977)
# Sentinel Chicken WN Seroconversion 1978 to 2015 Time Series
SCWNPF78to15ts<- ts(SCWNPF78to15, start = 1978, end = 2015, frequency = 1)
# Sentinel Chicken WN Seroconversion 1978 to 2013
SCWNPF78to13<- subset(bbs$SCWNPF, bbs$Year>1977 & bbs$Year<2014)
# Sentinel Chicken WN Seroconversion 1978 to 2013 Time Series
SCWNPF78to13ts<- ts(SCWNPF78to13, start = 1978, end = 2013, frequency = 1)
# Sentinel Chicken WN Seroconversion 2001 to 2015
SCWNPF01to15<- subset(bbs$SCWNPF, bbs$Year>2000)
# Sentinel Chicken WN Seroconversion 2001 to 2015 Time Series
SCWNPF01to15ts<- ts(SCWNPF01to15, start = 2001, end = 2015, frequency = 1)
# Sentinel Chicken WN Seroconversion 1978 to 2015
SCFlaviPF78to15<- subset(bbs$SCFlaviPF, bbs$Year>1977)
# Sentinel Chicken Flavi Seroconversion 1978 to 2015 Time Series
SCFlaviPF78to15ts<- ts(SCFlaviPF78to15, start = 1978, end = 2015, frequency = 1)
# Sentinel Chicken Flavi Seroconversion 1978 to 2013
SCFlaviPF78to13<- subset(bbs$SCFlaviPF, bbs$Year>1977 & bbs$Year<2014)
# Sentinel Chicken Flavi Seroconversion 1978 to 2013 Time Series
SCFlaviPF78to13ts<- ts(SCFlaviPF78to13, start = 1978, end = 2013, frequency = 1)
# Sentinel Chicken Flavi Seroconversion 2001 to 2015
SCFlaviPF01to15<- subset(bbs$SCFlaviPF, bbs$Year>2000)
# Sentinel Chicken Flavi Seroconversion 2001 to 2015 Time Series
SCFlaviPF01to15ts<- ts(SCFlaviPF01to15, start = 2001, end = 2015, frequency = 1)

# Human SLE/WN Variables
# Human SLE Cases 1958 to 2015
HumSLEPF58to15<- bbs$HumSLEPF
# Human SLE Cases 1958 to 2015 Time Series
HumSLEPF58to15ts<- ts(HumSLEPF58to15, start = 1958, end = 2015, frequency = 1)
# Human SLE Cases 1966 to 2015
HumSLEPF66to15<- subset(bbs$HumSLEPF, bbs$Year>1965)
# Human SLE Cases 1966 to 2015 Time Series
HumSLEPF66to15ts<- ts(HumSLEPF66to15, start = 1966, end = 2015, frequency = 1)
# Human SLE Cases 1966 to 2013
HumSLEPF66to13<- subset(bbs$HumSLEPF, bbs$Year>1965 & bbs$Year<2014)
# Human SLE Cases 1966 to 2013 Time Series
HumSLEPF66to13ts<- ts(HumSLEPF66to13, start = 1966, end = 2013, frequency = 1)
# Human SLE Cases 1978 to 2015
HumSLEPF78to15<- subset(bbs$HumSLEPF, bbs$Year>1977)
# Human SLE Cases 1978 to 2015 Time Series
HumSLEPF78to15ts<- ts(HumSLEPF78to15, start = 1978, end = 2015, frequency = 1)
# Human SLE Cases 2001 to 2015
HumSLEPF01to15<- subset(bbs$HumSLEPF, bbs$Year>2000)
# Human SLE Cases 2001 to 2015 Time Series
HumSLEPF01to15ts<- ts(HumSLEPF01to15, start = 2001, end = 2015, frequency = 1)
# Human WN Cases 2001 to 2015
HumWNPF01to15<- subset(bbs$HumSLEPF, bbs$Year>2000)
# Human WN Cases 2001 to 2015 Time Series
HumWNPF01to15ts<- ts(HumWNPF01to15, start = 2001, end = 2015, frequency = 1)
```

Preliminary Data Analysis

Summarize Data

Data Summary

```r
# Dataset
pander(bbs)
```

Table continues below

Year	RoutesPF	BJCountPF	CGCountPF	MDCountPF	NCCountPF	NMCountPF
1958	NA	NA	NA	NA	NA	NA
1959	NA	NA	NA	NA	NA	NA
1960	NA	NA	NA	NA	NA	NA
1961	NA	NA	NA	NA	NA	NA
1962	NA	NA	NA	NA	NA	NA
1963	NA	NA	NA	NA	NA	NA
1964	NA	NA	NA	NA	NA	NA
1965	NA	NA	NA	NA	NA	NA
1966	13	265	470	114	626	737
1967	17	328	617	212	617	1009
1968	17	325	708	285	852	1328
1969	18	439	917	359	960	1330
1970	21	478	1073	322	1078	1400
1971	22	420	920	468	1084	1602
1972	21	482	918	458	891	1682
1973	21	461	1027	479	849	1727
1974	15	294	837	467	636	1002
1975	16	374	748	503	514	1370
1976	18	362	834	534	585	1459
1977	18	456	863	692	616	1565
1978	17	339	704	482	555	1332
1979	17	409	823	476	514	1300
1980	16	361	778	482	445	1178
1981	15	270	693	412	455	1036
1982	14	261	636	369	448	967
1983	14	260	592	395	390	753
1984	11	217	569	245	233	621
1985	11	251	611	380	349	658
1986	18	401	825	882	514	1087
1987	50	845	1775	1638	1361	2446
1988	47	882	1987	1862	1448	2469
1989	42	807	1477	2022	1282	2189
1990	40	847	1514	1881	1279	2273
1991	42	774	1462	1668	1334	2148
1992	40	710	1281	2047	1448	2106
1993	44	920	1343	2040	1593	2099
1994	51	957	1484	2525	1610	2438
1995	43	789	1116	2275	1405	1913
1996	41	791	974	1757	1438	1641
1997	38	954	778	2180	1427	1783
1998	43	870	900	2003	1584	1757
1999	47	947	1184	2545	1784	2004
2000	46	918	1050	2345	1723	1718
2001	46	825	836	2465	1910	1852
2002	52	1014	1416	2615	2012	1960
2003	54	675	1067	2178	2059	1934
2004	54	861	1002	2138	2023	1745
2005	50	737	882	2025	2089	1576
2006	52	764	1056	2029	1995	1548
2007	52	673	985	1890	2041	1459

2008	46	586	876	1899	1782	1371
2009	47	606	902	1534	1887	1326
2010	53	640	931	2042	2185	1410
2011	53	705	892	2130	2225	1493
2012	47	646	685	2177	1972	1223
2013	47	611	796	1985	1889	1169
2014	46	483	857	1470	1704	1002
2015	41	394	637	1366	1488	844

Table continues below

BJAveragePF	CGAveragePF	MDAveragePF	NCAveragePF	NMAveragePF
NA	NA	NA	NA	NA
NA	NA	NA	NA	NA
NA	NA	NA	NA	NA
NA	NA	NA	NA	NA
NA	NA	NA	NA	NA
NA	NA	NA	NA	NA
NA	NA	NA	NA	NA
NA	NA	NA	NA	NA
30.22	15.55	8.769	40.25	18.31
26.3	23.46	12.47	26.3	38.36
19.39	36.52	16.77	23.33	56.92
22.01	41.66	19.94	23.04	57.72
31.17	34.42	15.33	31.32	44.7
19.74	46.6	21.27	23.26	68.86
22.1	41.54	21.81	21.45	78.41
20.21	50.81	22.81	16.71	103.4
9.443	88.64	31.13	7.176	139.6
11.9	62.88	31.44	8.175	167.6
12.2	68.35	29.67	8.559	170.5
11.86	72.76	38.44	8.466	184.8
11.96	58.88	28.35	9.426	141.3
14.61	56.34	28	9.123	142.5
11.98	64.92	30.12	6.854	171.9
9.83	70.5	27.47	6.454	160.5
9.902	64.23	26.36	6.975	138.6
9.215	64.24	28.21	6.071	124
9.743	58.4	22.27	3.99	155.7
7.266	84.09	34.55	4.15	158.5
8.184	100.8	49	5.099	213.2
25.79	68.81	32.76	19.78	123.7
22.26	89.25	39.62	16.22	152.2
16.76	88.11	48.14	14.55	150.5
18.01	84.06	47.02	15.22	149.4
19.49	75.02	39.71	17.78	120.8
13.87	92.33	51.17	15.68	134.3
19.84	67.68	46.36	23.54	89.18
19.33	76.77	49.51	20.97	116.3
14.91	74.83	52.91	18.77	101.9
18.46	52.77	42.85	27.25	60.22
16.63	46.78	57.37	30.5	58.46
18.68	48.19	46.58	32.87	53.45
17.49	67.7	54.15	26.35	76.05

18.01	58.31	50.98	29.55	58.14
15.4	54.3	53.59	35.17	52.65
20.16	70.22	50.29	28.65	68.41
16.74	63.76	40.33	32.3	59.89
21.75	46.08	39.59	43.91	39.74
18.2	48.47	40.5	43.1	36.57
19.58	53.93	39.02	36.99	41.85
18.52	53.2	36.35	38.37	38.03
14.2	61.71	41.28	28.88	47.48
18.57	48.58	32.64	38.84	34.14
16.61	56.05	38.53	38.98	36.17
17.54	50.85	40.19	43.76	34.12
13.95	49.12	46.32	40.15	30.46
14.47	55.02	42.23	34.33	34.05
15.11	56.7	31.96	30.05	33.34
11.83	53.87	33.32	27.62	30.55

Table continues below

BJIndexPF	CGIndexPF	MDIndexPF	NCIndexPF	NMIndexPF	SCFlaviPF	SCWNPF
NA	NA	NA	NA	NA	NA	NA
NA	NA	NA	NA	NA	NA	NA
NA	NA	NA	NA	NA	NA	NA
NA	NA	NA	NA	NA	NA	NA
NA	NA	NA	NA	NA	NA	NA
NA	NA	NA	NA	NA	NA	NA
NA	NA	NA	NA	NA	NA	NA
NA	NA	NA	NA	NA	NA	NA
20.09	70.92	20.76	40.68	112.7	NA	NA
19.19	64.05	17.63	40.64	103.9	NA	NA
20.05	64.86	19.47	44.46	108.9	NA	NA
20.21	72.42	21.4	44.94	102.4	NA	NA
19.49	71.92	17.76	45.89	98.87	NA	NA
18.9	63.07	23.51	46.87	99.37	NA	NA
19.03	59.62	22.71	43.46	101.2	NA	NA
18.45	72.37	22.93	40.7	98.35	NA	NA
17.85	64.4	25.74	42.52	94.36	NA	NA
17.7	56.86	27.54	39.78	93.95	NA	NA
17.35	56.93	26.56	39.27	92.2	NA	NA
18.29	56.03	32.27	38.77	90.46	NA	NA
16.59	48.41	29.09	39.28	84.43	0	NA
17.44	52	28.27	40.42	83.78	147	NA
17.11	52.03	29.1	38.43	80.57	105	NA
16.59	51.76	28.15	38.62	78.77	35	NA
16.86	49.5	29.96	37.91	76.41	143	NA
16.18	46.36	32.44	38.62	71.99	169	NA
15.91	51.55	29.09	35.33	69.96	124	NA
16.19	47.12	34.21	39.87	68.66	212	NA
16.13	48.09	37.93	37.37	67.58	182	NA
15.36	46.16	35	38	67.77	10	NA
15.8	44.99	35.65	40.59	66.98	6	NA
16.09	37.48	40.56	38.94	65.09	141	NA
15.76	39.81	38.63	40.79	64.36	1023	NA
15.03	44.82	44.79	40.05	64.77	80	NA

14.87	45.6	46.77	40.54	63.48	4	NA
15.35	41.93	47.27	39.32	59.88	218	NA
14.92	37.08	51.56	41.19	58.97	84	NA
13.76	32.55	46.18	39.73	54.55	54	NA
13.91	28.16	40.83	40.6	51.07	21	NA
14.97	27.4	48.66	42.61	53.24	628	NA
13.36	25.47	44.49	39.85	50.94	213	NA
14.01	27.7	52.07	40.97	49.22	193	NA
13.38	22.17	49.91	39.53	46.64	155	NA
13.31	22.73	55.39	42.29	48.95	109	90
14.28	27.12	54.98	41.73	48.44	1050	1050
11.19	17.57	48.53	40.21	46.51	1008	997
12.93	17.57	45.27	39.03	42.83	289	278
12.16	17.88	42.42	41.14	40.7	331	326
11.47	18.06	45.1	41.32	39.28	53	13
11.11	18.08	45.32	41.85	38.58	20	18
11.14	19.85	49.13	39.98	38.52	7	7
10.67	16.56	42.39	43.28	36.82	55	55
10.84	14.4	50.2	43.24	36.21	499	494
11.25	17.52	54.21	43.39	36.82	249	185
11.24	12.54	61.27	42.46	35.79	308	226
10.98	16.48	57.97	43.61	34.8	223	135
NA	NA	NA	NA	NA	389	291
NA	NA	NA	NA	NA	410	401

SCSLEPF	HumSLEPF	Row
NA	5	1
NA	70	2
NA	0	3
NA	25	4
NA	222	5
NA	0	6
NA	0	7
NA	0	8
NA	0	9
NA	0	10
NA	0	11
NA	3	12
NA	0	13
NA	0	14
NA	0	15
NA	0	16
NA	0	17
NA	0	18
NA	0	19
NA	138	20
0	0	21
147	6	22
105	1	23
35	1	24
143	1	25
169	0	26
124	0	27
212	1	28

182	0	29
10	0	30
6	0	31
141	0	32
1023	223	33
80	0	34
4	0	35
218	8	36
84	0	37
54	0	38
21	0	39
628	9	40
213	2	41
193	3	42
155	0	43
19	0	44
0	0	45
11	0	46
11	0	47
5	0	48
40	0	49
2	0	50
0	0	51
0	0	52
5	0	53
64	0	54
82	0	55
88	0	56
98	0	57
9	0	58

```r
# Summarize BBS and Transmission Data
summary(bbs)
```

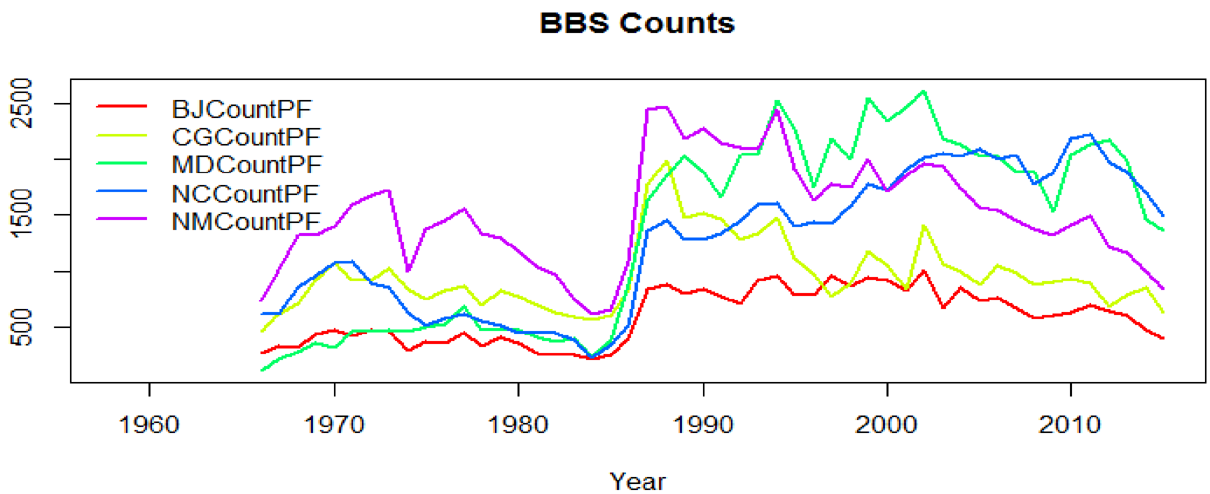
```
      Year         RoutesPF       BJCountPF        CGCountPF        MDCountPF
 Min.   :1958   Min.   :11.00   Min.   : 217.0   Min.   : 470.0   Min.   : 114
 1st Qu.:1972   1st Qu.:17.25   1st Qu.: 379.0   1st Qu.: 778.0   1st Qu.: 470
 Median :1986   Median :41.00   Median : 608.5   Median : 896.0   Median :1653
 Mean   :1986   Mean   :34.08   Mean   : 593.7   Mean   : 966.2   Mean   :1355
 3rd Qu.:2001   3rd Qu.:47.00   3rd Qu.: 803.0   3rd Qu.:1064.2   3rd Qu.:2042
 Max.   :2015   Max.   :54.00   Max.   :1014.0   Max.   :1987.0   Max.   :2615
                NA's   :8       NA's   :8        NA's   :8        NA's   :8
   NCCountPF         NMCountPF      BJAveragePF      CGAveragePF      MDAveragePF
 Min.   : 233.0   Min.   : 621   Min.   : 7.266   Min.   : 15.55   Min.   : 8.769
 1st Qu.: 619.2   1st Qu.:1189   1st Qu.:12.620   1st Qu.: 49.54   1st Qu.:28.249
 Median :1383.0   Median :1476   Median :17.126   Median : 58.35   Median :38.486
 Mean   :1263.8   Mean   :1521   Mean   :16.828   Mean   : 60.36   Mean   :36.189
 3rd Qu.:1783.5   3rd Qu.:1835   3rd Qu.:19.557   3rd Qu.: 69.87   3rd Qu.:46.353
 Max.   :2225.0   Max.   :2469   Max.   :31.174   Max.   :100.81   Max.   :57.368
 NA's   :8        NA's   :8      NA's   :8        NA's   :8        NA's   :8
  NCAveragePF     NMAveragePF       BJIndexPF       CGIndexPF       MDIndexPF
 Min.   : 3.99   Min.   : 18.31   Min.   :10.67   Min.   :12.54   Min.   :17.63
 1st Qu.:10.71   1st Qu.: 42.56   1st Qu.:13.35   1st Qu.:22.59   1st Qu.:28.24
 Median :23.30   Median : 72.46   Median :15.56   Median :44.91   Median :39.59
 Mean   :22.93   Mean   : 91.95   Mean   :15.31   Mean   :40.79   Mean   :38.15
 3rd Qu.:32.05   3rd Qu.:140.90   3rd Qu.:17.37   3rd Qu.:53.03   3rd Qu.:47.59
 Max.   :43.91   Max.   :213.19   Max.   :20.21   Max.   :72.42   Max.   :61.27
 NA's   :8       NA's   :8        NA's   :10      NA's   :10      NA's   :10
   NCIndexPF       NMIndexPF        SCFlaviPF          SCWNPF          SCSLEPF
 Min.   :35.33   Min.   : 34.80   Min.   :   0.00   Min.   :   7.0   Min.   :   0.00
 1st Qu.:39.31   1st Qu.: 47.99   1st Qu.:  54.25   1st Qu.:  72.5   1st Qu.:   9.25
 Median :40.59   Median : 64.93   Median : 151.00   Median : 226.0   Median :  72.00
 Mean   :40.83   Mean   : 67.08   Mean   : 235.45   Mean   : 304.4   Mean   : 115.29
```

```
3rd Qu.:42.33   3rd Qu.: 85.94   3rd Qu.: 279.00   3rd Qu.: 363.5   3rd Qu.: 146.00
Max.   :46.87   Max.   :112.70   Max.   :1050.00   Max.   :1050.0   Max.   :1023.00
NA's   :10      NA's   :10       NA's   :20        NA's   :43       NA's   :20
   HumSLEPF          Row
Min.   :  0.00   Min.   : 1.00
1st Qu.:  0.00   1st Qu.:15.25
Median :  0.00   Median :29.50
Mean   : 12.38   Mean   :29.50
3rd Qu.:  1.00   3rd Qu.:43.75
Max.   :223.00   Max.   :58.00
```

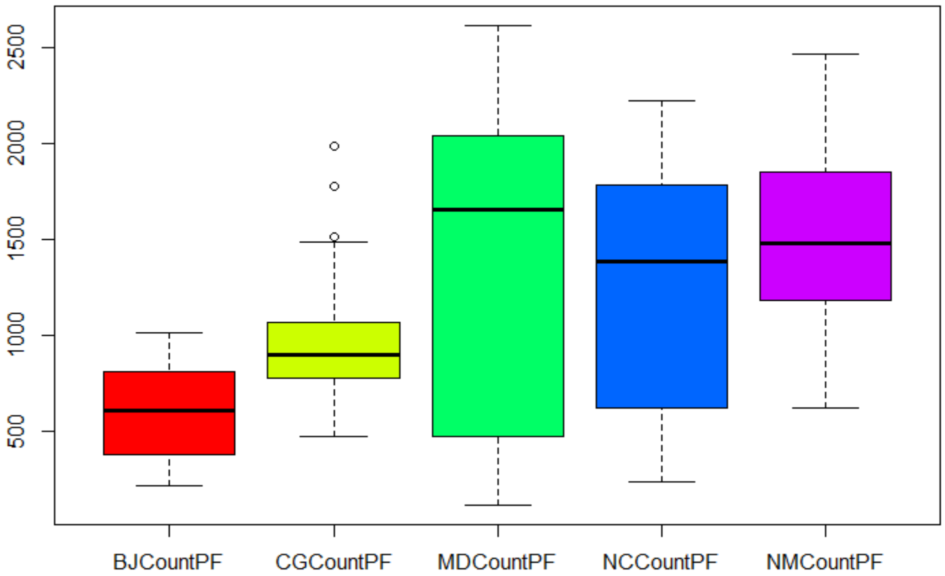
Plot Data

Plot BBS Data

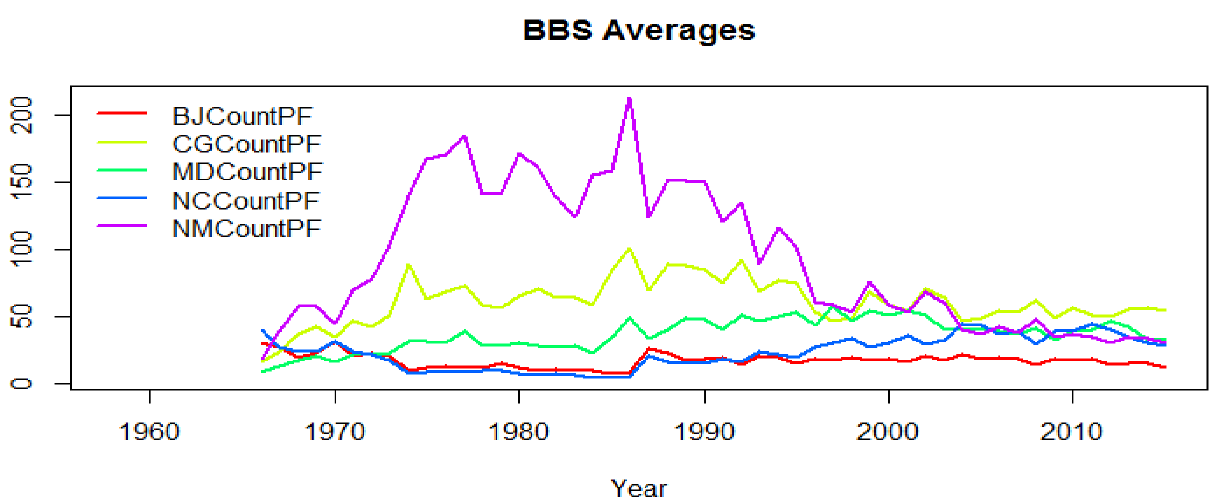
```r
# Plot BBS Counts
ts.plot(bbs58to15ts[,3:7],gpars= list(col=rainbow(5)),lty=1,lwd=2, main="BBS Counts",
         xlab="Year")
legend("topleft", colnames(bbs58to15ts[,3:7]), lty=1, col=rainbow(5), bty="n", cex=1,
        lwd=2)
```



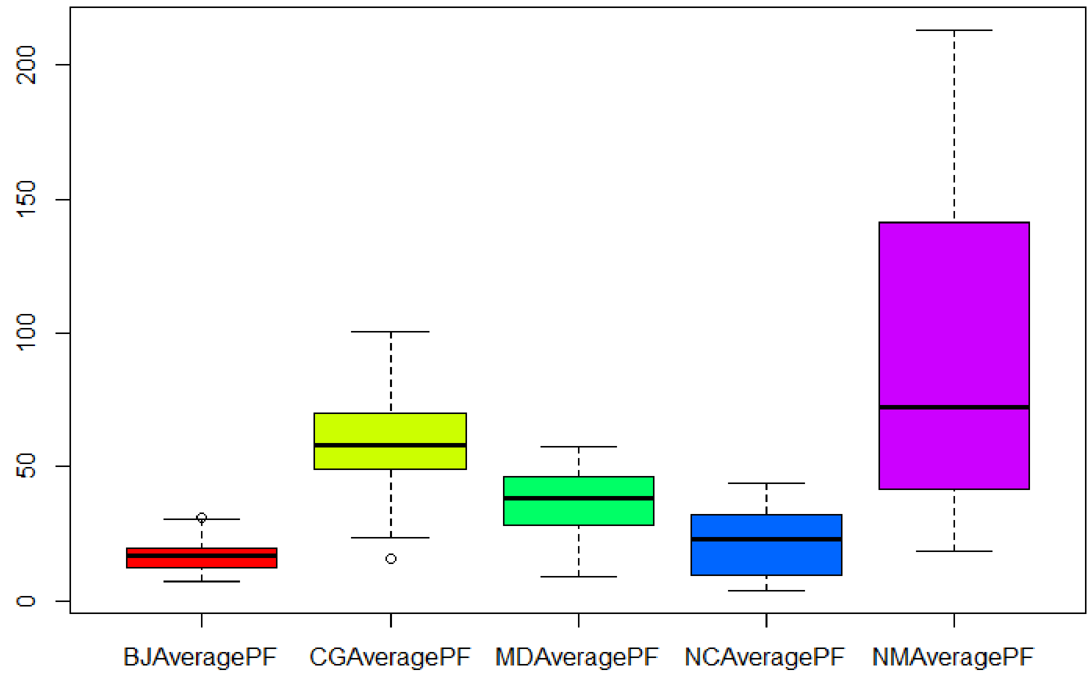

```r
# BBS Counts Boxplot
boxplot(bbs58to15ts[,3:7], col=rainbow(5))
```



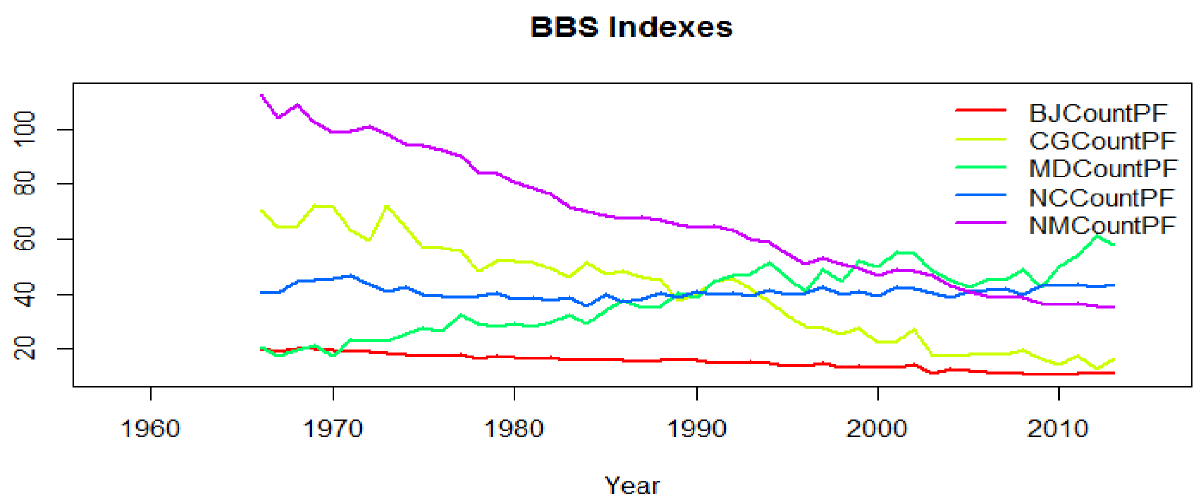

```
# Plot BBS Averages
ts.plot(bbs58to15ts[,8:12],gpars= list(col=rainbow(5)),lty=1,lwd=2,
              main="BBS Averages", xlab="Year")
legend("topleft", colnames(bbs58to15ts[,3:7]), lty=1, col=rainbow(5),
            bty="n", cex=1, lwd=2)
```



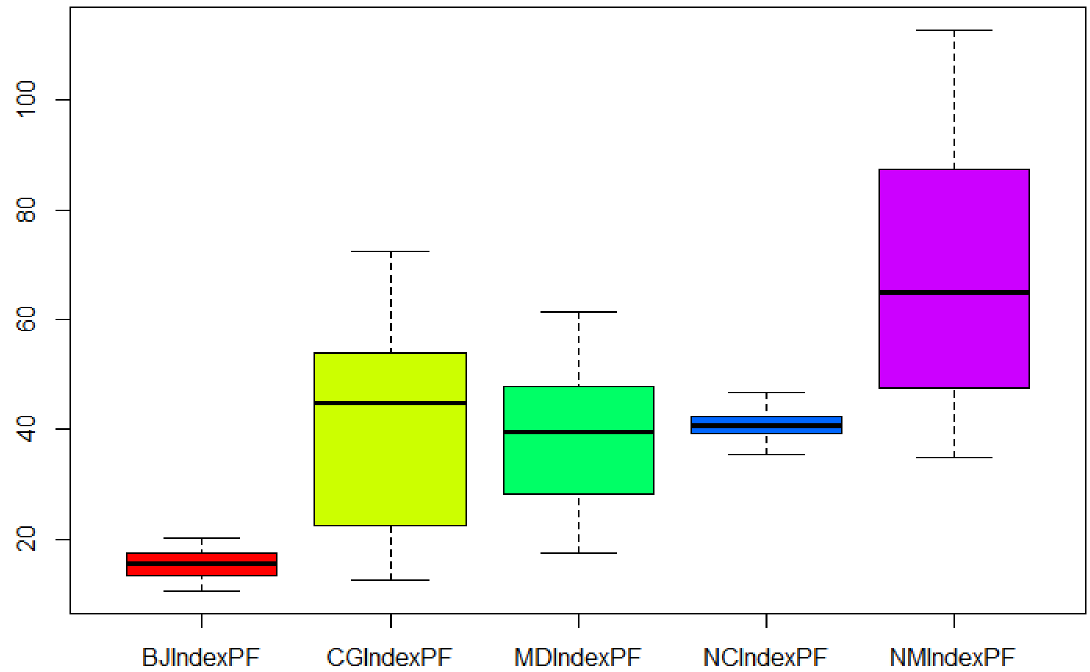

```
# BBS Counts Boxplot
boxplot(bbs58to15ts[,8:12], col=rainbow(5))
```



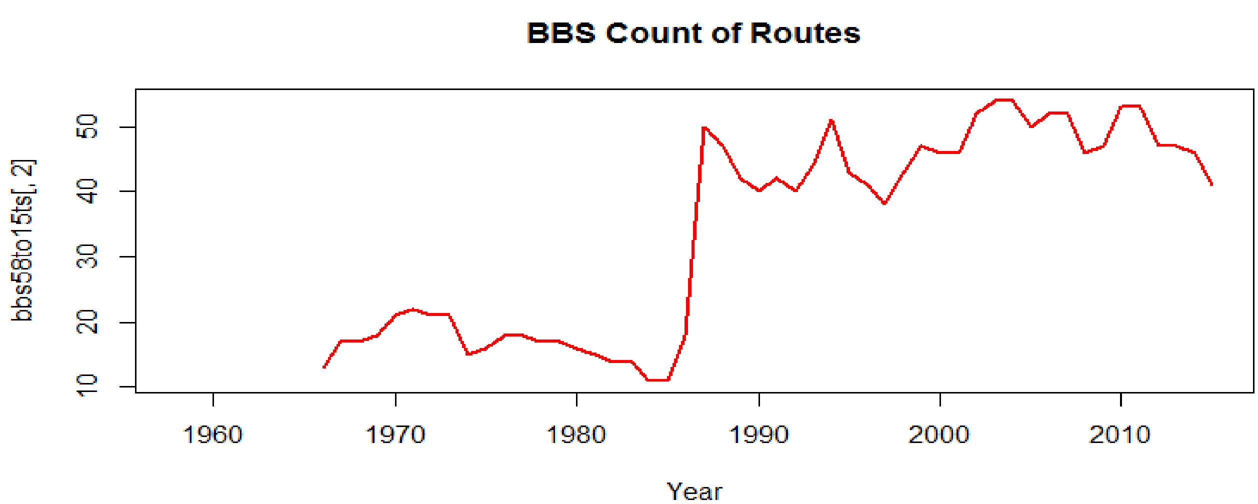

```r
# Plot BBS Index
ts.plot(bbs58to15ts[,13:17],gpars= list(col=rainbow(5)),lty=1,lwd=2,
        main="BBS Indexes", xlab="Year")
legend("topright", colnames(bbs58to15ts[,3:7]), lty=1, col=rainbow(5),
       bty="n", cex=1, lwd=2)
```




```r
# BBS Index Boxplot
boxplot(bbs58to15ts[,13:17], col=rainbow(5))
```

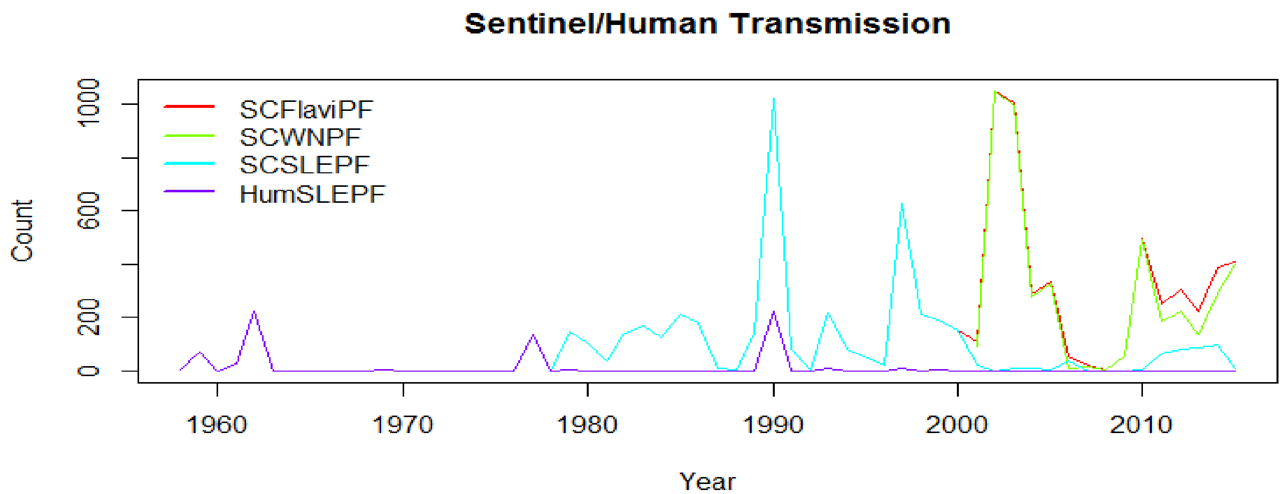



```
# Plot BBS Count of Routes
ts.plot(bbs58to15ts[,2],gpars= list(col=rainbow(5)),lty=1,lwd=2,
        main="BBS Count of Routes", xlab="Year")
```

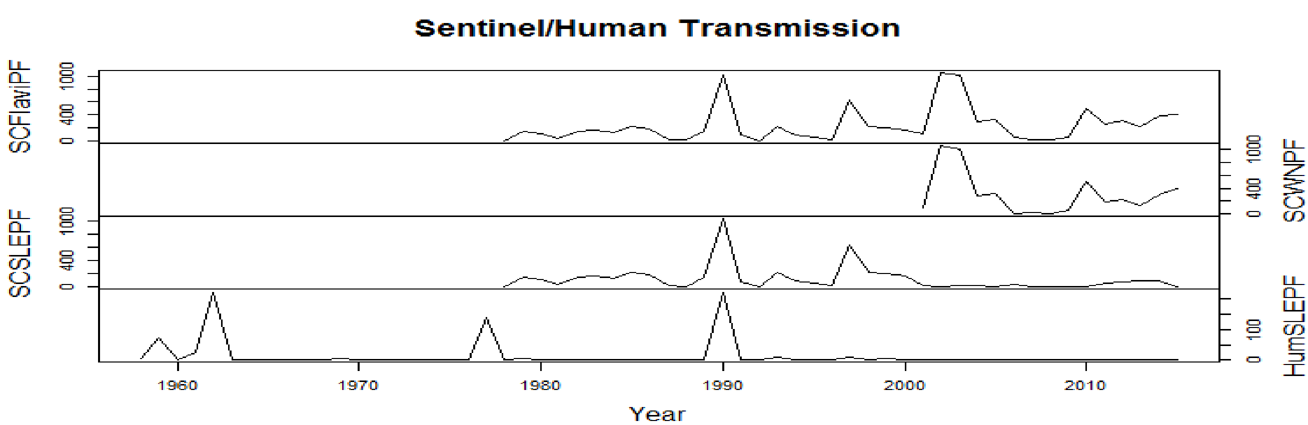


Plot Transmission Data

```
# Plot Transmission Counts
ts.plot(bbs58to15ts[,18:21], main="Sentinel/Human Transmission",
        xlab="Year", ylab="Count",  gpars= list(col=rainbow(4)))
legend("topleft", colnames(bbs58to15ts[,18:21]), lty=1, col=rainbow(4),
       bty="n", cex=1, lwd=2)
```




```
plot.ts(bbs58to15ts[,18:21], main="Sentinel/Human Transmission", xlab="Year",
        yax.flip=TRUE)
```



Check Data Distribution

BBS Count Data Distribution

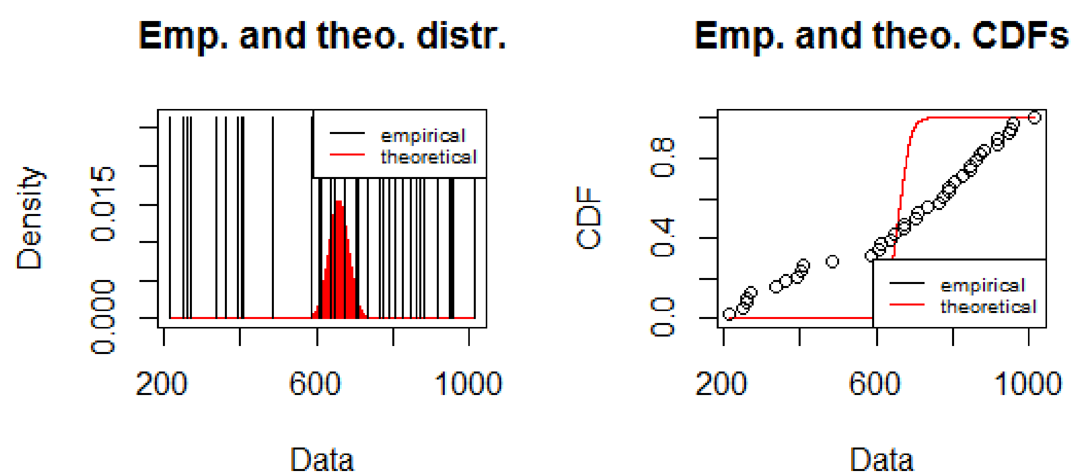
Note: Data has been filtered from the time period of 1966-2015 to 1978-2015 to match Transmission data.

Blue Jay Count Data

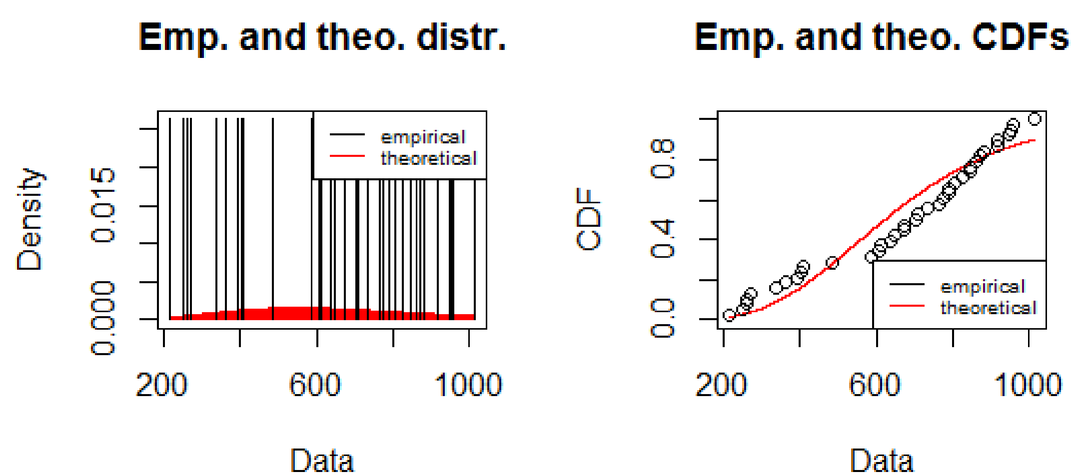
```
gofstat(list(fitdist(BJCountPF78to15, "pois"), fitdist(BJCountPF78to15,
        "nbinom")),fitnames = c("Poisson","negative binomial"))

Chi-squared statistic:  1.701847e+65 16.67283
Degree of freedom of the Chi-squared distribution:  5 4
Chi-squared p-value:  0 0.002237288
   the p-value may be wrong with some theoretical counts < 5
Chi-squared table:
       obscounts theo Poisson theo negative binomial
<= 270         5 1.468992e-64               1.475122
<= 409         5 4.281669e-24               5.086137
<= 640         5 9.497877e+00              13.490714
<= 710         5 2.769906e+01               3.810044
<= 791         5 8.030560e-01               3.763373
<= 861         5 8.293156e-06               2.636885
> 861          8 6.581402e-13               7.737726
```

```

Goodness-of-fit criteria
                                Poisson negative binomial
Aikake's Information Criterion 3887.792          532.1100
Bayesian Information Criterion 3889.430          535.3851
```

```
plot(fitdist(BJCountPF78to15, "pois"))
```




```
plot(fitdist(BJCountPF78to15, "nbinom"))
```

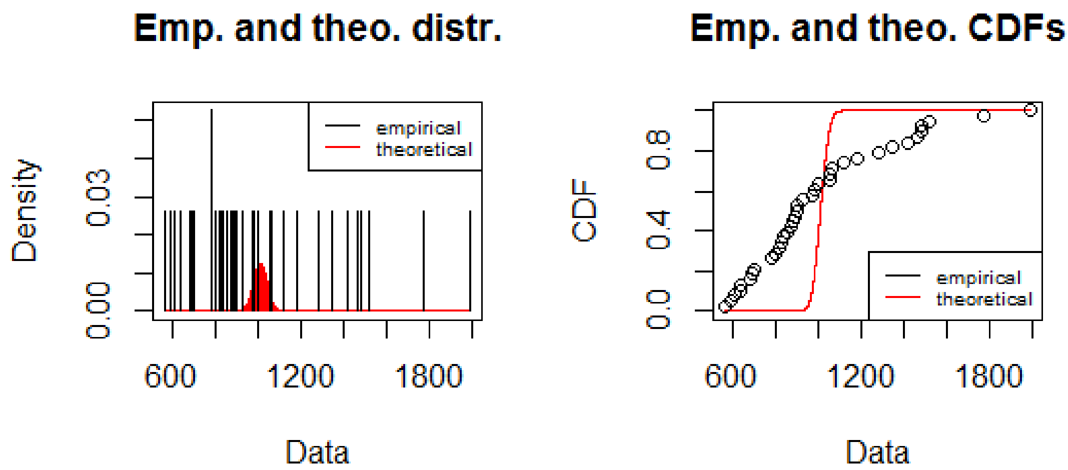


Common Grackle Count Data

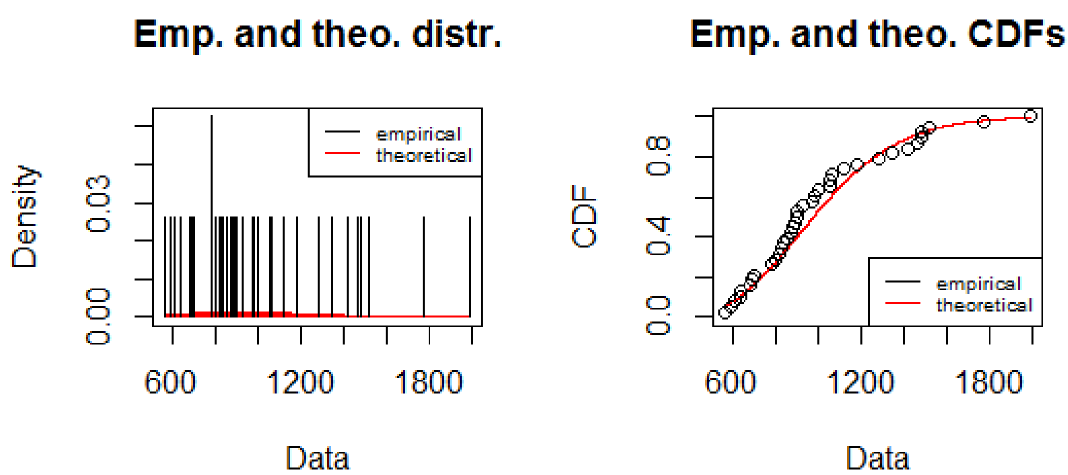
```
gofstat(list(fitdist(CGCountPF78to15, "pois"), fitdist(CGCountPF78to15,
        "nbinom")),fitnames = c("Poisson","negative binomial"))

Chi-squared statistic:  4.44328e+35 6.304185
Degree of freedom of the Chi-squared distribution:  5 4
Chi-squared p-value:  0 0.1775541
   the p-value may be wrong with some theoretical counts < 5
Chi-squared table:
        obscounts theo Poisson theo negative binomial
<= 637          5 5.626474e-35              4.048452
<= 778          5 6.341631e-13              5.344474
<= 857          5 1.652901e-05              3.722909
```

```
<= 902          5 1.116675e-02              2.222108
<= 1050         5 3.414180e+01              7.124919
<= 1281         5 3.847012e+00              8.436609
> 1281          8 4.218847e-15              7.100530

Goodness-of-fit criteria
                                Poisson negative binomial
Aikake's Information Criterion 4320.188          547.3024
Bayesian Information Criterion 4321.826          550.5776
```

```
plot(fitdist(CGCountPF78to15, "pois"))
```




```
plot(fitdist(CGCountPF78to15, "nbinom"))
```

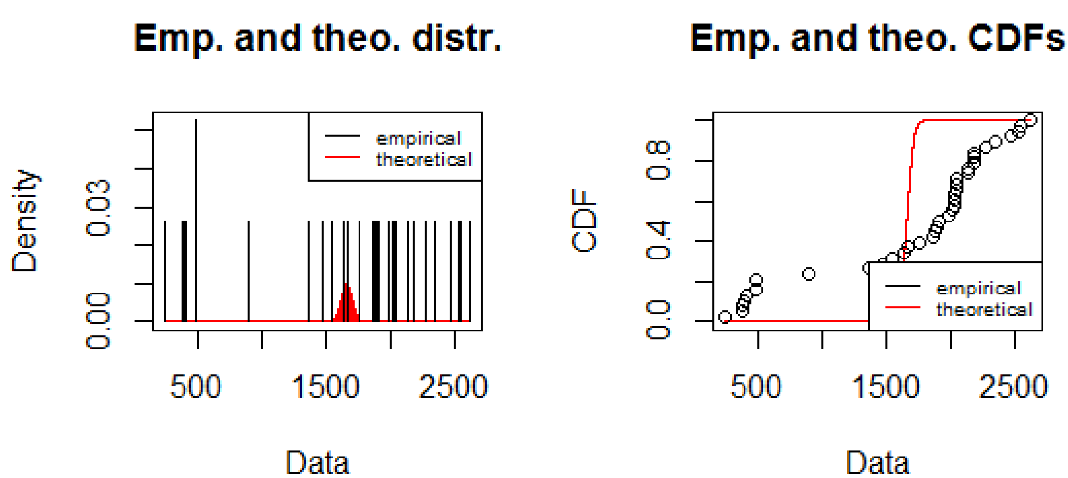


Mourning Dove Count Data

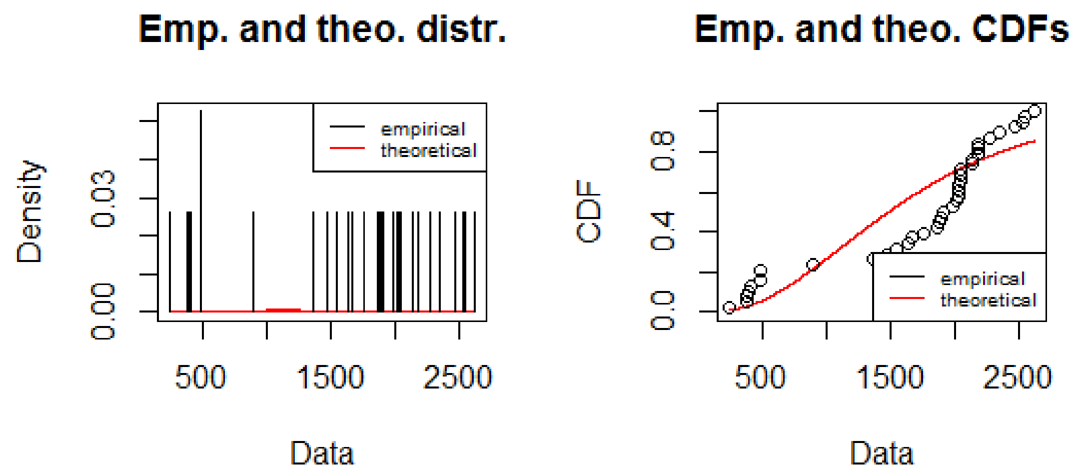
```
gofstat(list(fitdist(MDCountPF78to15, "pois"), fitdist(MDCountPF78to15,
   "nbinom")),fitnames = c("Poisson","negative binomial"))
```

```
Chi-squared statistic:  Inf 53.85901
Degree of freedom of the Chi-squared distribution:  5 4
Chi-squared p-value:  0 5.632849e-11
   the p-value may be wrong with some theoretical counts < 5
Chi-squared table:
        obscounts  theo Poisson theo negative binomial
<= 412          5 1.929814e-291              1.4462055
<= 1366         5  6.001212e-12             15.5732262
<= 1757         5  3.777816e+01              6.4213815
<= 1985         5  2.218443e-01              3.0768532
<= 2040         5  5.062617e-14              0.6638387
<= 2177         5  0.000000e+00              1.5241102
> 2177          8  0.000000e+00              9.2943848

Goodness-of-fit criteria
                                Poisson negative binomial
Aikake's Information Criterion 15391.23          623.4892
Bayesian Information Criterion 15392.87          626.7643
```

```
plot(fitdist(MDCountPF78to15, "pois"))
```




```
plot(fitdist(MDCountPF78to15, "nbinom"))
```

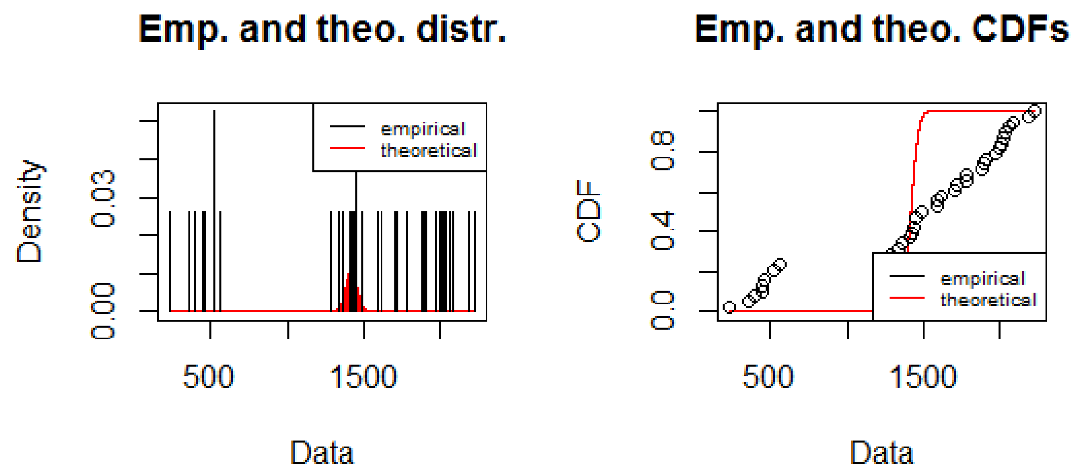


Northern Cardinal Count Data

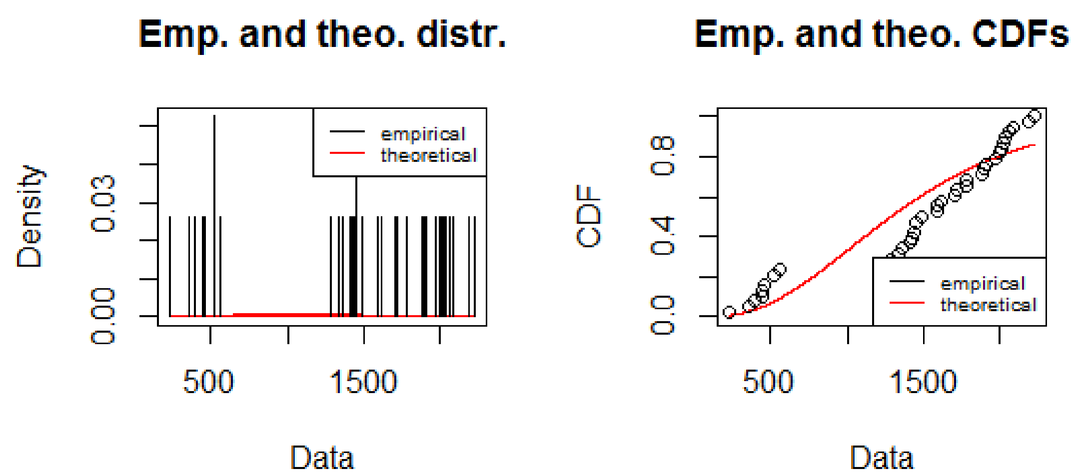
```
gofstat(list(fitdist(NCCountPF78to15, "pois"), fitdist(NCCountPF78to15,
    "nbinom")),fitnames = c("Poisson","negative binomial"))
```

```
Chi-squared statistic:  Inf 19.66904
Degree of freedom of the Chi-squared distribution:  5 4
Chi-squared p-value:  0 0.0005804062
   the p-value may be wrong with some theoretical counts < 5
Chi-squared table:
        obscounts  theo Poisson theo negative binomial
<= 448          5 9.198400e-198             2.001736
<= 1279         5  3.592193e-03            16.865214
<= 1427         5  2.288257e+01             3.020797
<= 1584         5  1.511357e+01             2.873381
<= 1782         5  2.631727e-04             3.090476
<= 1972         5  0.000000e+00             2.413790
> 1972          8  0.000000e+00             7.734606

Goodness-of-fit criteria
                                Poisson negative binomial
Aikake's Information Criterion 12202.38          607.9049
Bayesian Information Criterion 12204.01          611.1801
```

```
plot(fitdist(NCCountPF78to15, "pois"))
```



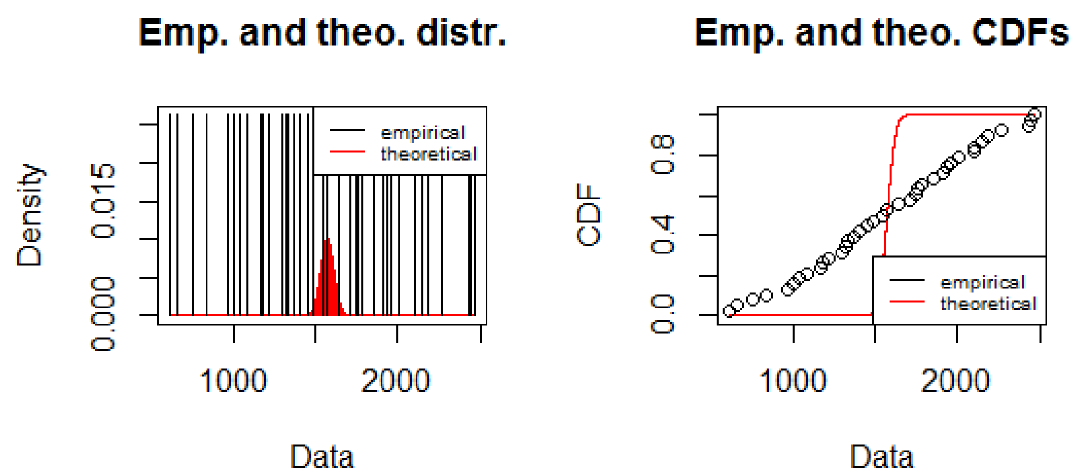

```
plot(fitdist(NCCountPF78to15, "nbinom"))
```



Northern Mockingbird Count Data

```
gofstat(list(fitdist(NMCountPF78to15, "pois"), fitdist(NMCountPF78to15,
        "nbinom")),fitnames = c("Poisson","negative binomial"))

Chi-squared statistic:  Inf 0.4396675
Degree of freedom of the Chi-squared distribution:  5 4
Chi-squared p-value:  0 0.9791023
   the p-value may be wrong with some theoretical counts < 5
Chi-squared table:
        obscounts theo Poisson theo negative binomial
<= 967          5 1.021725e-59             4.354095
<= 1178         5 2.996759e-24             4.912066
<= 1371         5 3.291339e-06             5.533249
<= 1576         5 1.985732e+01             5.975441
<= 1783         5 1.814267e+01             5.309187
<= 2004         5 4.623188e-06             4.395894
> 2004          8 0.000000e+00             7.520069

Goodness-of-fit criteria
                                Poisson negative binomial
```

```
Aikake's Information Criterion 6825.623          586.5488
Bayesian Information Criterion 6827.261          589.8239

plot(fitdist(NMCountPF78to15, "pois"))
```



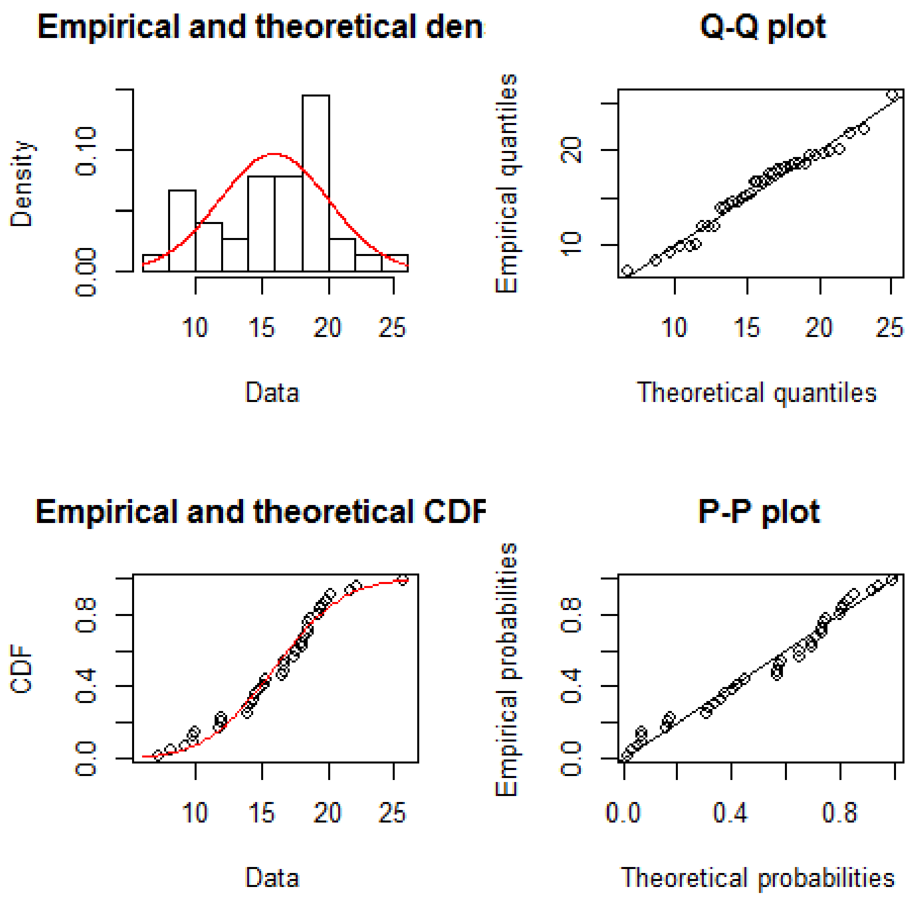

```
plot(fitdist(NMCountPF78to15, "nbinom"))
```

BBS Averages Data Distribution

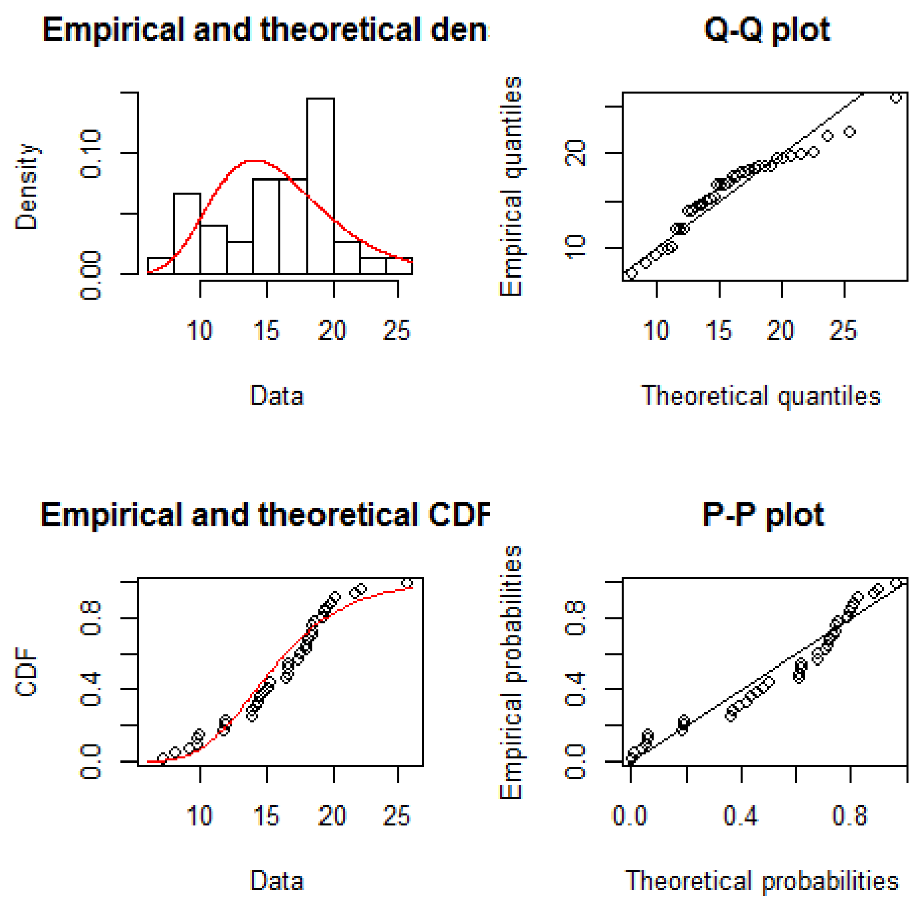
Blue Jay Average per Route Data

```
gofstat(list(fitdist(BJAveragePF78to15, "norm"), fitdist(BJAveragePF78to15,
        "lnorm")),fitnames = c("normal","lognormal"))

Goodness-of-fit statistics
                                 normal lognormal
Kolmogorov-Smirnov statistic 0.11939840 0.1633584
Cramer-von Mises statistic   0.07913803 0.1955069
Anderson-Darling statistic   0.48131266 1.1127937

Goodness-of-fit criteria
                                 normal lognormal
Aikake's Information Criterion 219.6162  224.7589
Bayesian Information Criterion 222.8914  228.0341
```

```
plot(fitdist(BJAveragePF78to15, "norm"))
```

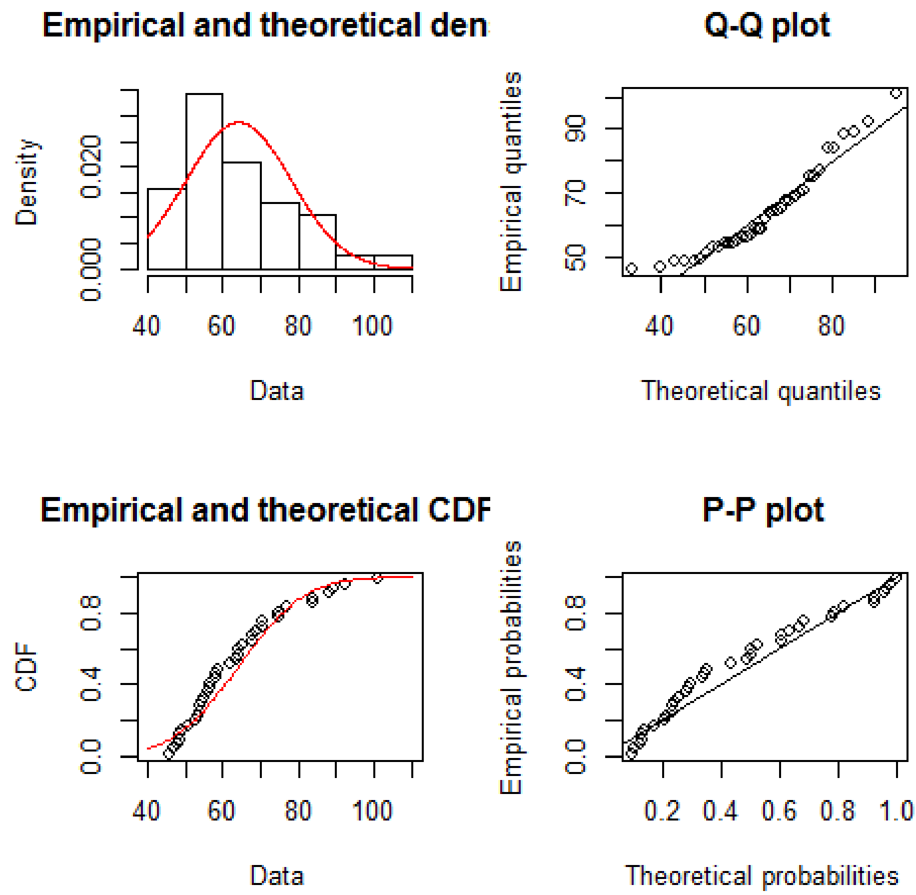



```
plot(fitdist(BJAveragePF78to15, "lnorm"))
```

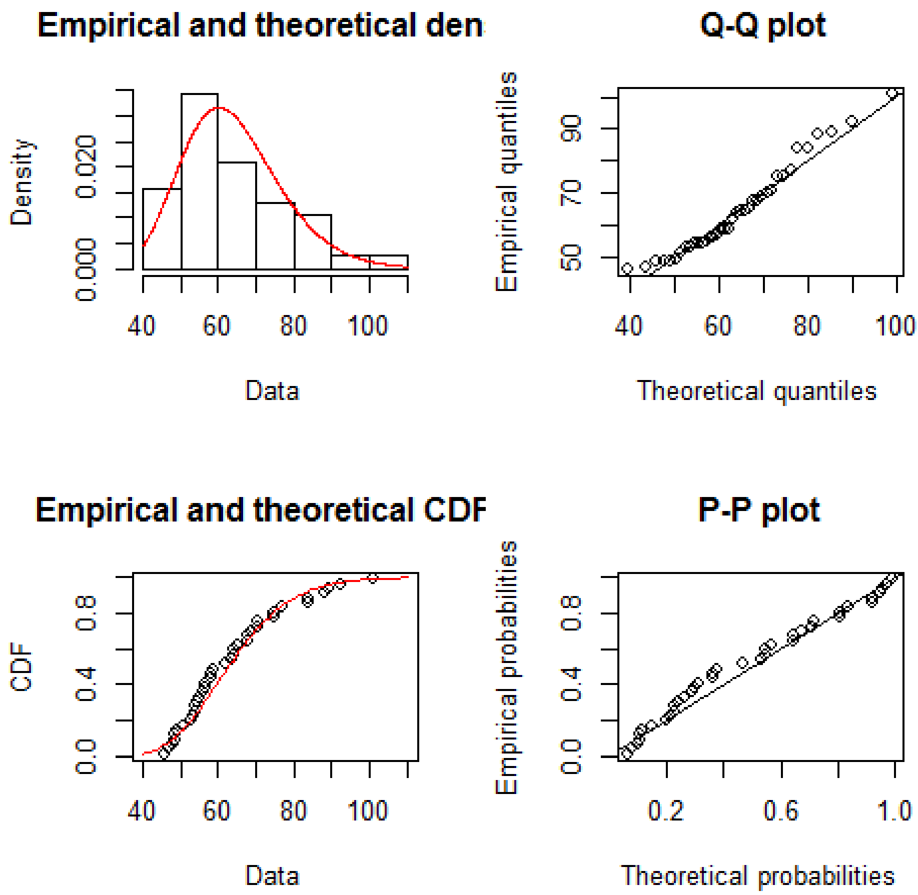


Common Grackle Average per Route Data

```
gofstat(list(fitdist(CGAveragePF78to15, "norm"), fitdist(CGAveragePF78to15,
        "lnorm")),fitnames = c("normal","lognormal"))

Goodness-of-fit statistics
                                normal  lognormal
Kolmogorov-Smirnov statistic 0.1456513 0.11933031
Cramer-von Mises statistic   0.1560498 0.08276156
Anderson-Darling statistic   0.9951086 0.55429903

Goodness-of-fit criteria
                                 normal lognormal
Aikake's Information Criterion 311.9038  306.3401
Bayesian Information Criterion 315.1790  309.6152

plot(fitdist(CGAveragePF78to15, "norm"))
```




```
plot(fitdist(CGAveragePF78to15, "lnorm"))
```

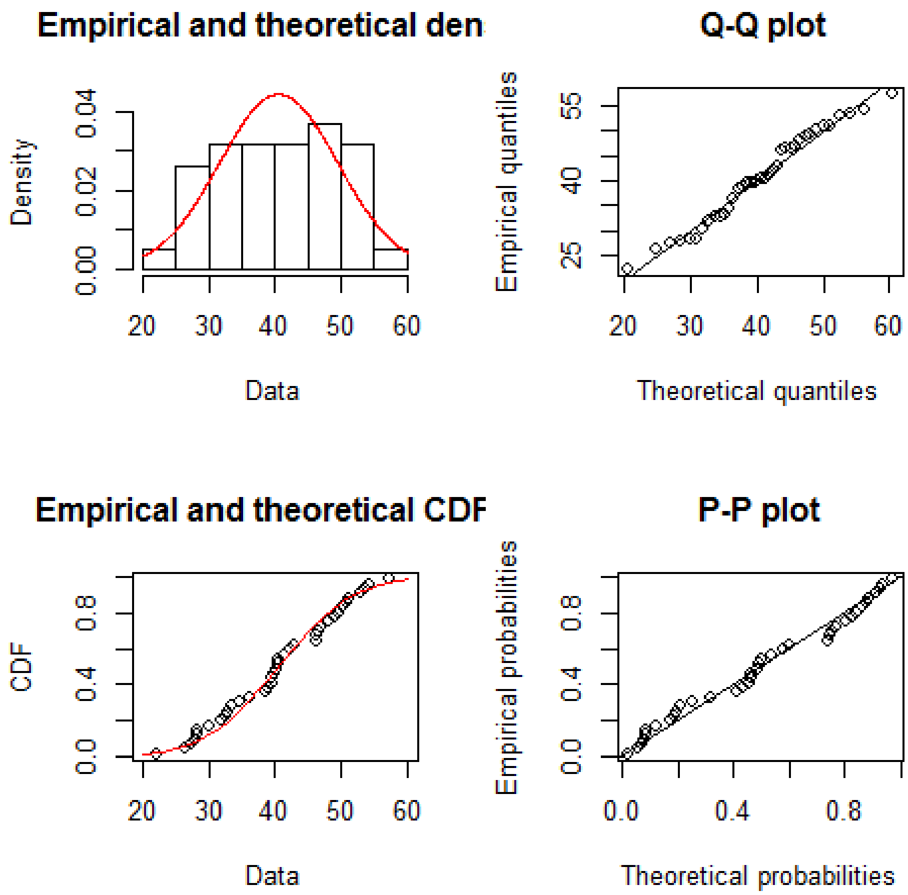


Mourning Dove Average per Route Data

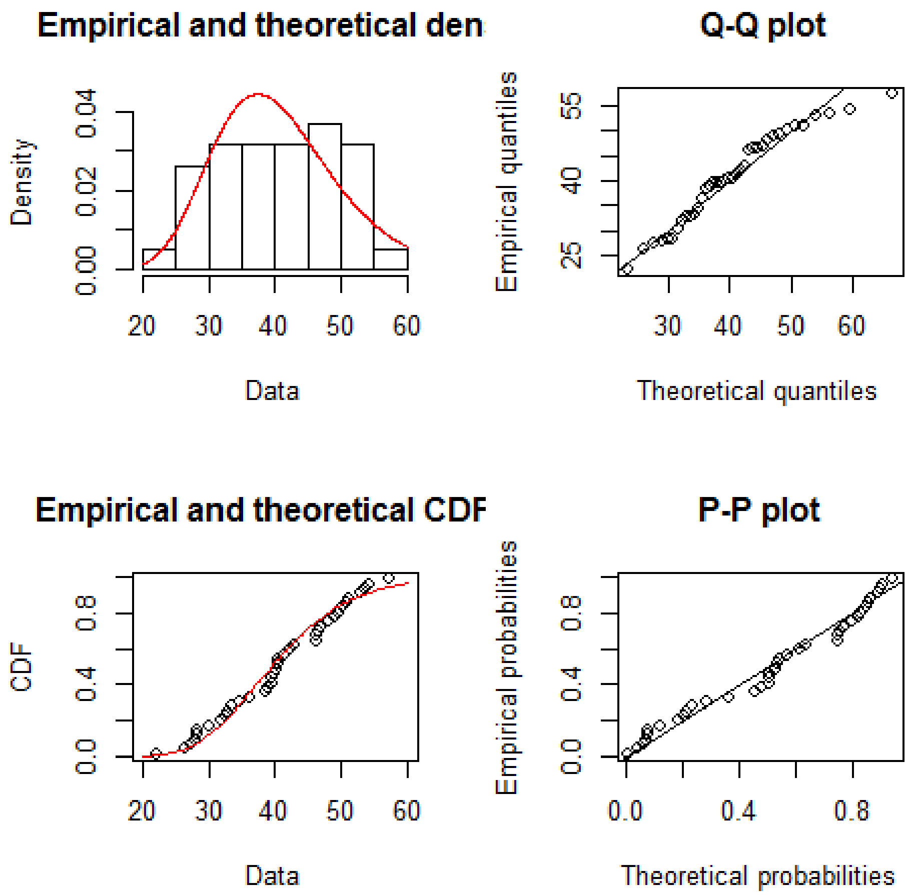
```
gofstat(list(fitdist(MDAveragePF78to15, "norm"), fitdist(MDAveragePF78to15,
        "lnorm")),fitnames = c("normal","lognormal"))
```

```
Goodness-of-fit statistics
                                 normal  lognormal
Kolmogorov-Smirnov statistic 0.10962314 0.12168779
Cramer-von Mises statistic   0.06282804 0.09443359
Anderson-Darling statistic   0.40472381 0.58905531

Goodness-of-fit criteria
                                 normal lognormal
Aikake's Information Criterion 278.5597  280.7233
Bayesian Information Criterion 281.8348  283.9984
```

```
plot(fitdist(MDAveragePF78to15, "norm"))
```

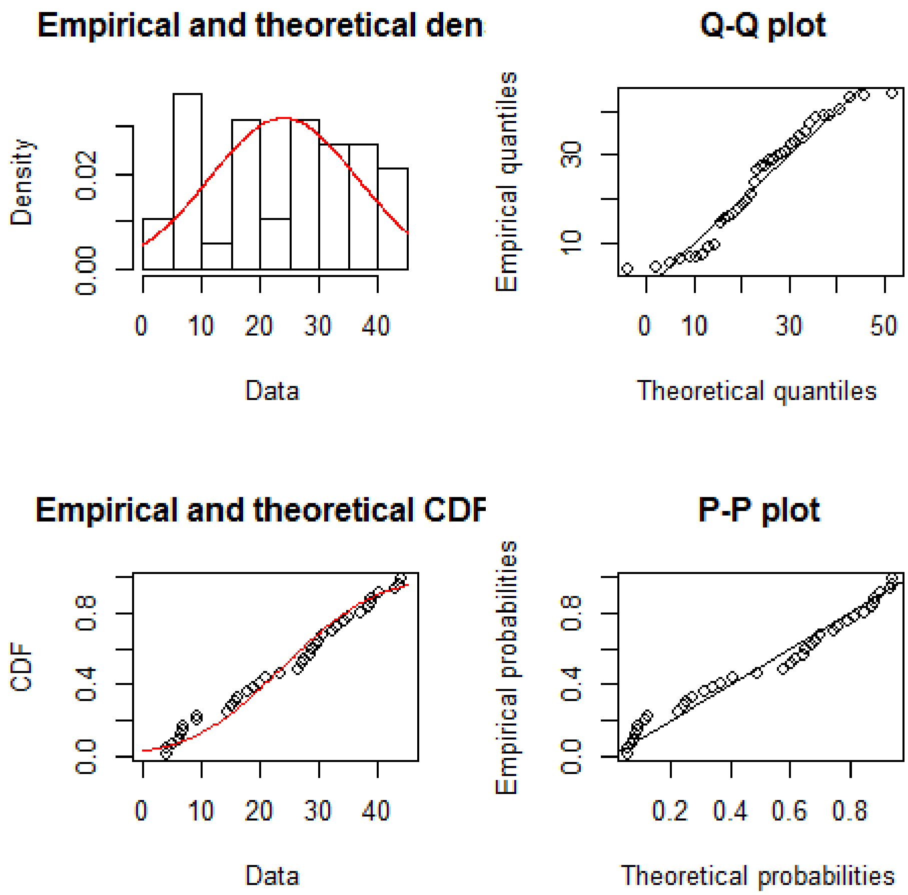



```
plot(fitdist(MDAveragePF78to15, "lnorm"))
```

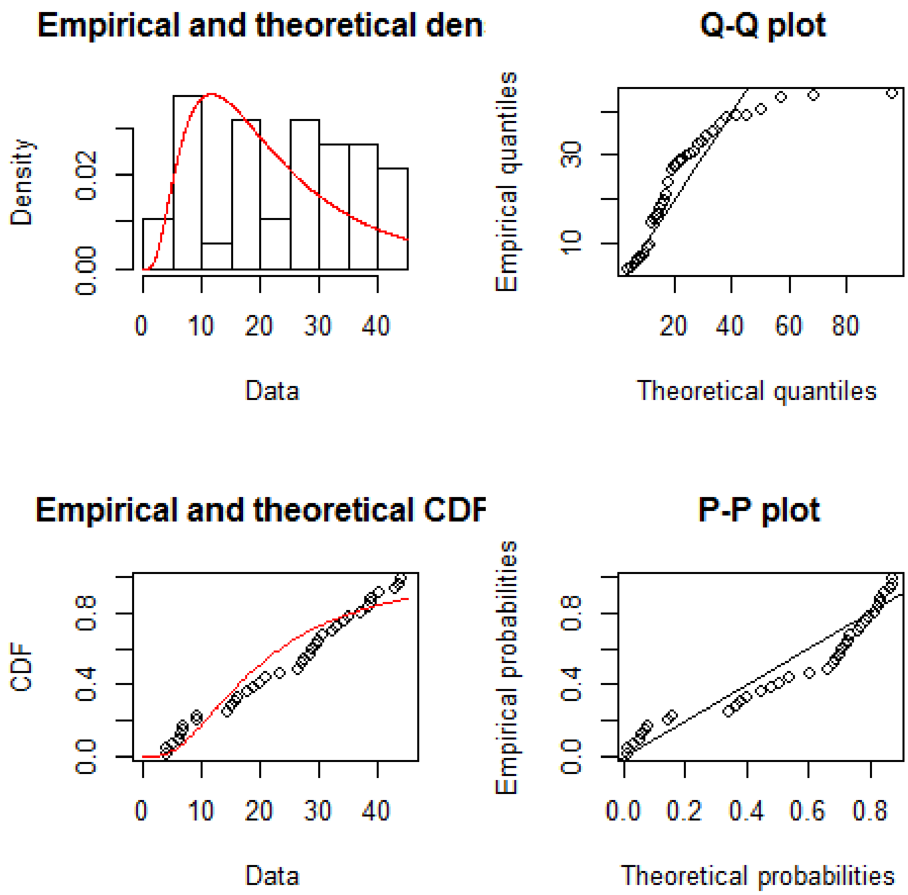


Northern Cardinal Average per Route Data

```
gofstat(list(fitdist(NCAveragePF78to15, "norm"), fitdist(NCAveragePF78to15,
        "lnorm")),fitnames = c("normal","lognormal"))

Goodness-of-fit statistics
                                 normal lognormal
Kolmogorov-Smirnov statistic 0.11252537 0.1896369
Cramer-von Mises statistic   0.09954104 0.2937678
Anderson-Darling statistic   0.69765544 1.7577813

Goodness-of-fit criteria
                                 normal lognormal
Aikake's Information Criterion 304.0857  312.1668
Bayesian Information Criterion 307.3609  315.4420

plot(fitdist(NCAveragePF78to15, "norm"))
```




```
plot(fitdist(NCAveragePF78to15, "lnorm"))
```

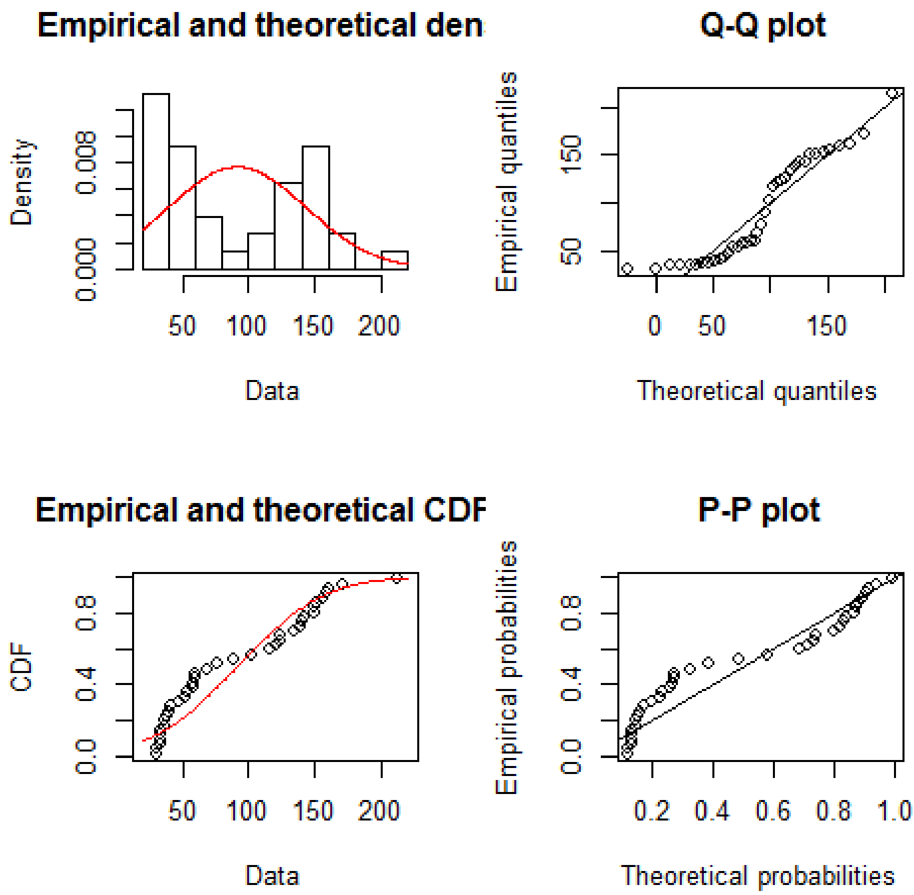


Northern Mockingbird Average per Route Data

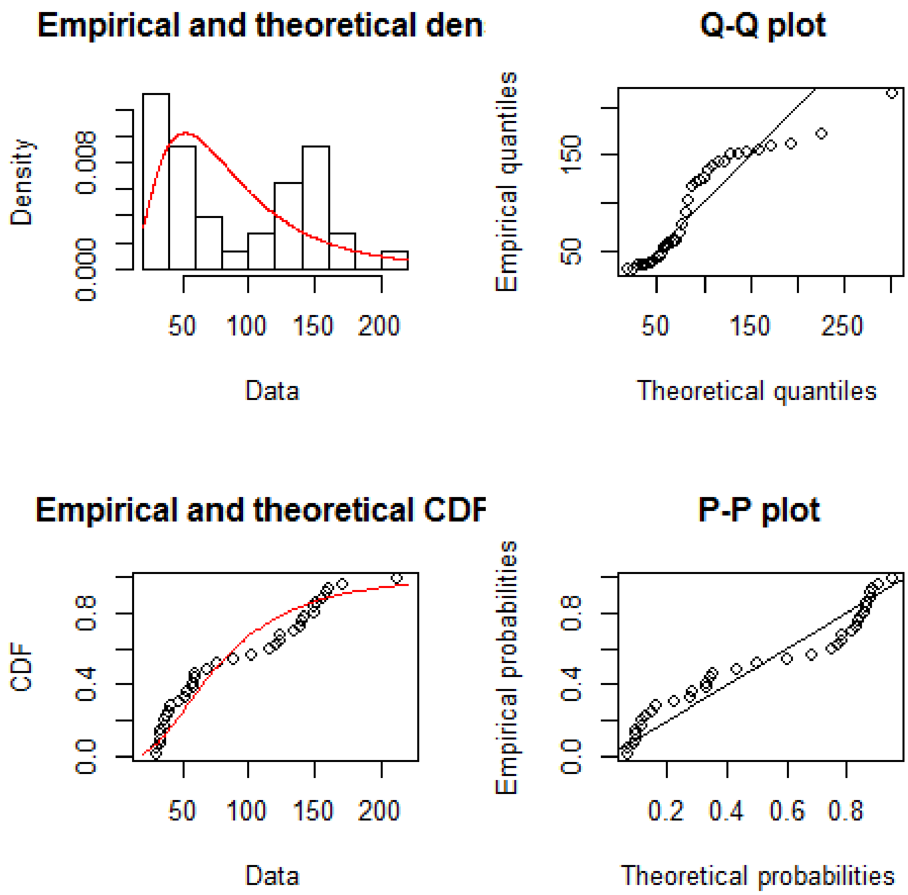
```
gofstat(list(fitdist(NMAveragePF78to15, "norm"), fitdist(NMAveragePF78to15,
        "lnorm")),fitnames = c("normal","lognormal"))
```

```
Goodness-of-fit statistics
                                normal lognormal
Kolmogorov-Smirnov statistic 0.1986121 0.1739295
Cramer-von Mises statistic   0.2974952 0.2459064
Anderson-Darling statistic   1.7330332 1.5334864

Goodness-of-fit criteria
                                normal lognormal
Aikake's Information Criterion 412.0772  404.6609
Bayesian Information Criterion 415.3524  407.9360
```

```
plot(fitdist(NMAveragePF78to15, "norm"))
```



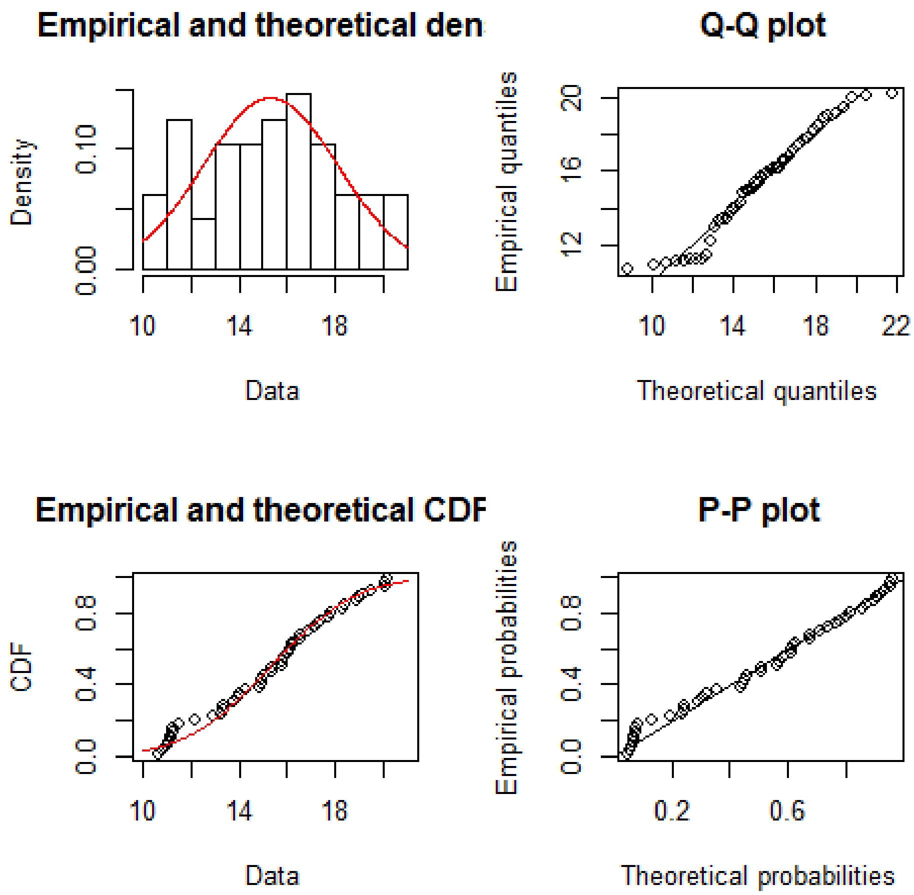

```
plot(fitdist(NMAveragePF78to15, "lnorm"))
```



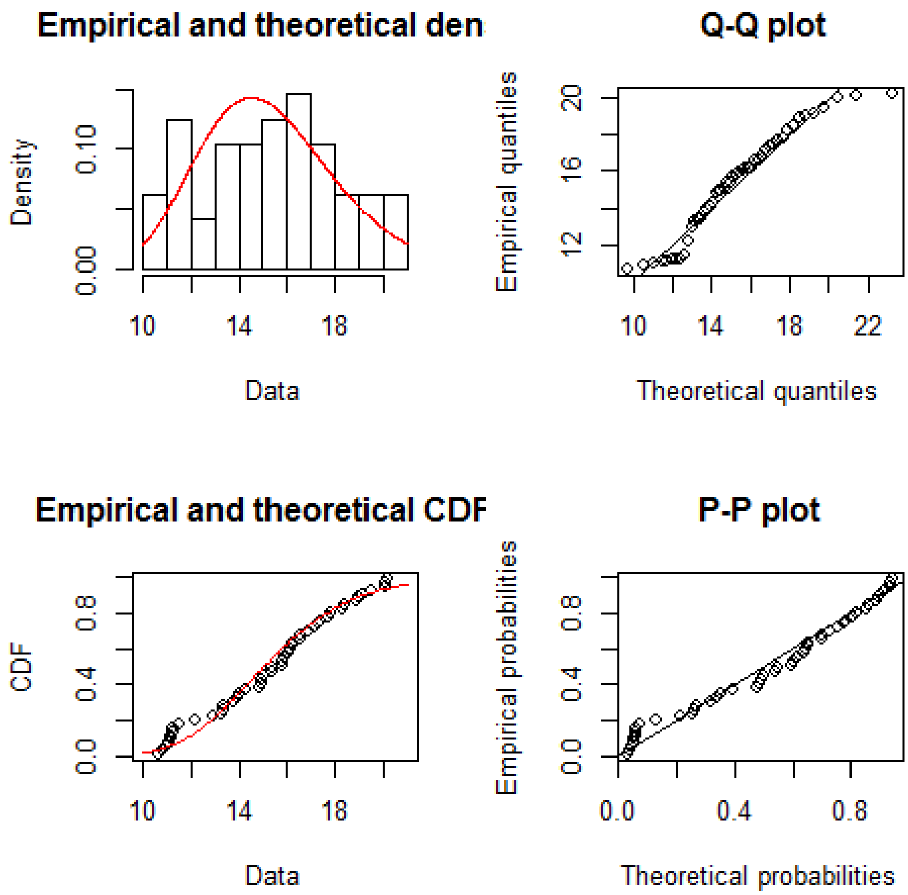
BBS Trend Data Distribution

Blue Jay Trend Index Data

```
gofstat(list(fitdist(BJIndexPF66to13, "norm"), fitdist(BJIndexPF66to13,
        "lnorm")),fitnames = c("normal","lognormal"))

Goodness-of-fit statistics
                                 normal lognormal
Kolmogorov-Smirnov statistic 0.10243763 0.1119496
Cramer-von Mises statistic   0.05528419 0.1046562
Anderson-Darling statistic   0.52425819 0.8182633

Goodness-of-fit criteria
                                 normal lognormal
Aikake's Information Criterion 238.9624  240.4658
Bayesian Information Criterion 242.7048  244.2082

plot(fitdist(BJIndexPF66to13, "norm"))
```

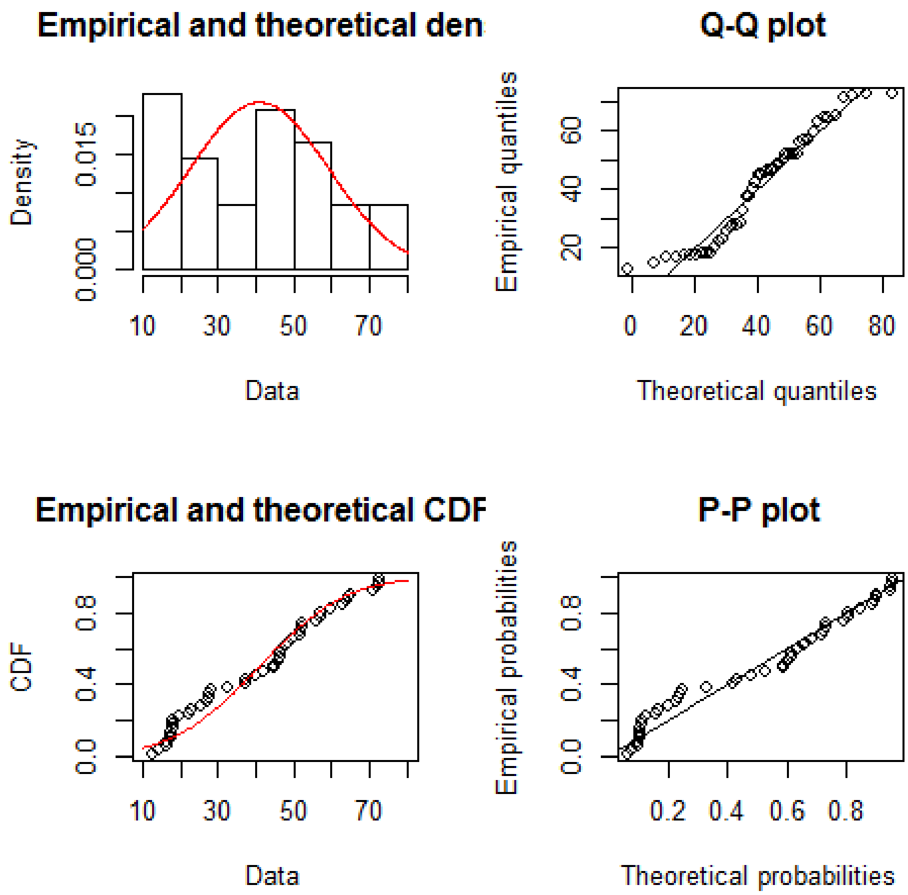



```
plot(fitdist(BJIndexPF66to13, "lnorm"))
```

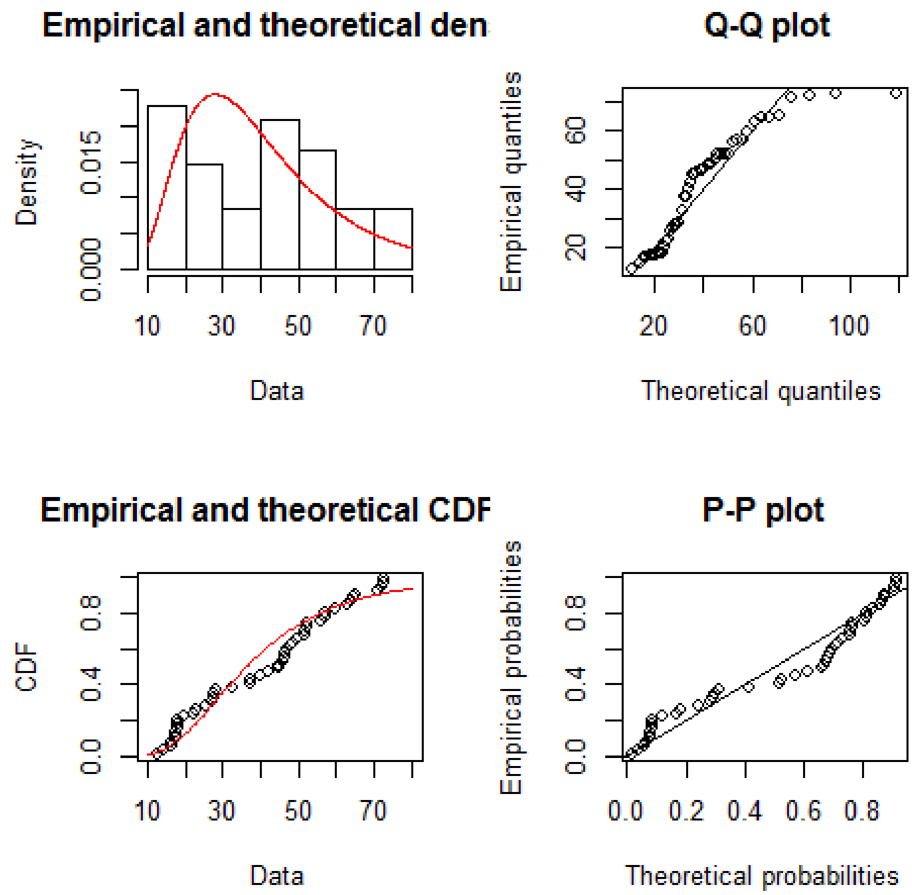


Common Grackle Trend Index Data

```
gofstat(list(fitdist(CGIndexPF66to13, "norm"), fitdist(CGIndexPF66to13,
        "lnorm")),fitnames = c("normal","lognormal"))

Goodness-of-fit statistics
                                normal lognormal
Kolmogorov-Smirnov statistic 0.1306454 0.1821228
Cramer-von Mises statistic   0.1494840 0.2801951
Anderson-Darling statistic   1.0099181 1.6223607

Goodness-of-fit criteria
                                 normal lognormal
Aikake's Information Criterion 418.9799  420.8320
Bayesian Information Criterion 422.7223  424.5744

plot(fitdist(CGIndexPF66to13, "norm"))
```




```
plot(fitdist(CGIndexPF66to13, "lnorm"))
```

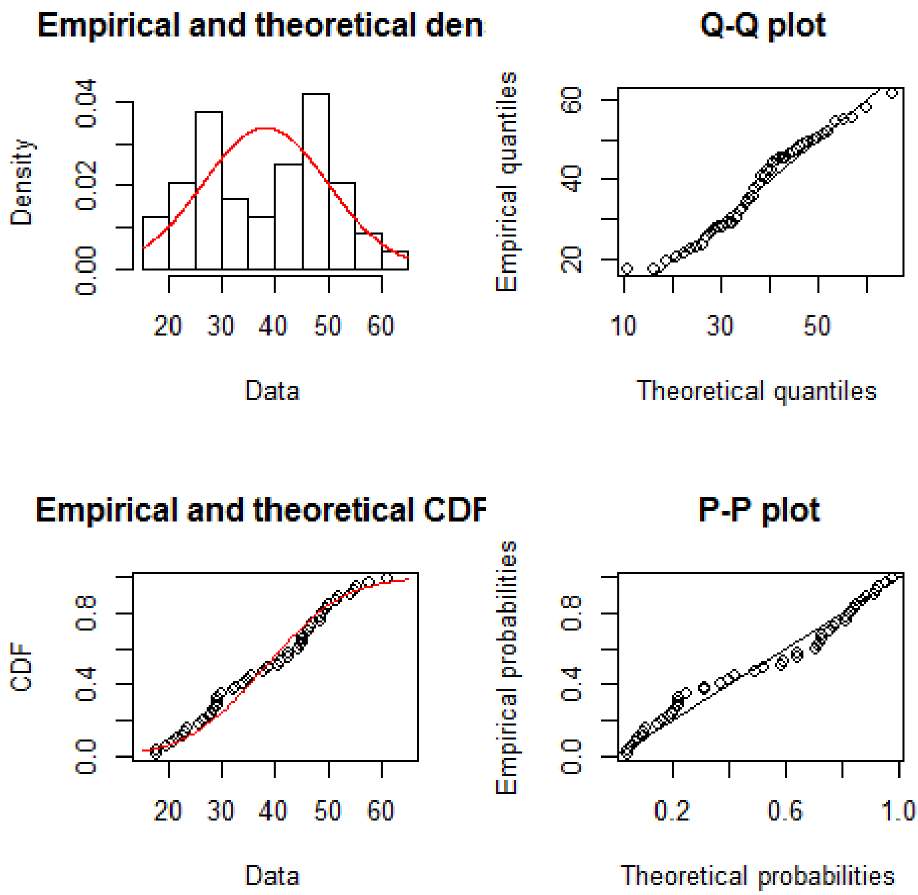


Mourning Dove Trend Index Data

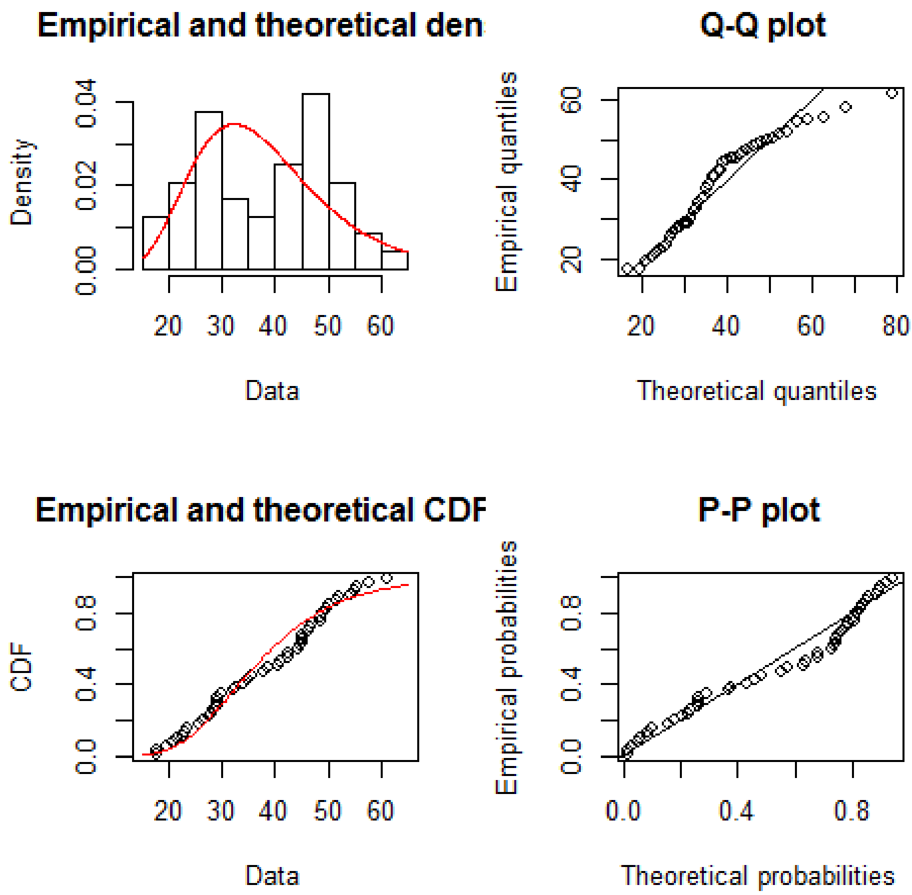
```
gofstat(list(fitdist(MDIndexPF66to13, "norm"), fitdist(MDIndexPF66to13,
        "lnorm")),fitnames = c("normal","lognormal"))
```

```
Goodness-of-fit statistics
                                normal lognormal
Kolmogorov-Smirnov statistic 0.1205500 0.1468991
Cramer-von Mises statistic   0.1267768 0.1768854
Anderson-Darling statistic   0.7075533 1.0204126

Goodness-of-fit criteria
                                 normal lognormal
Aikake's Information Criterion 377.4983  380.6026
Bayesian Information Criterion 381.2407  384.3450
```

```
plot(fitdist(MDIndexPF66to13, "norm"))
```

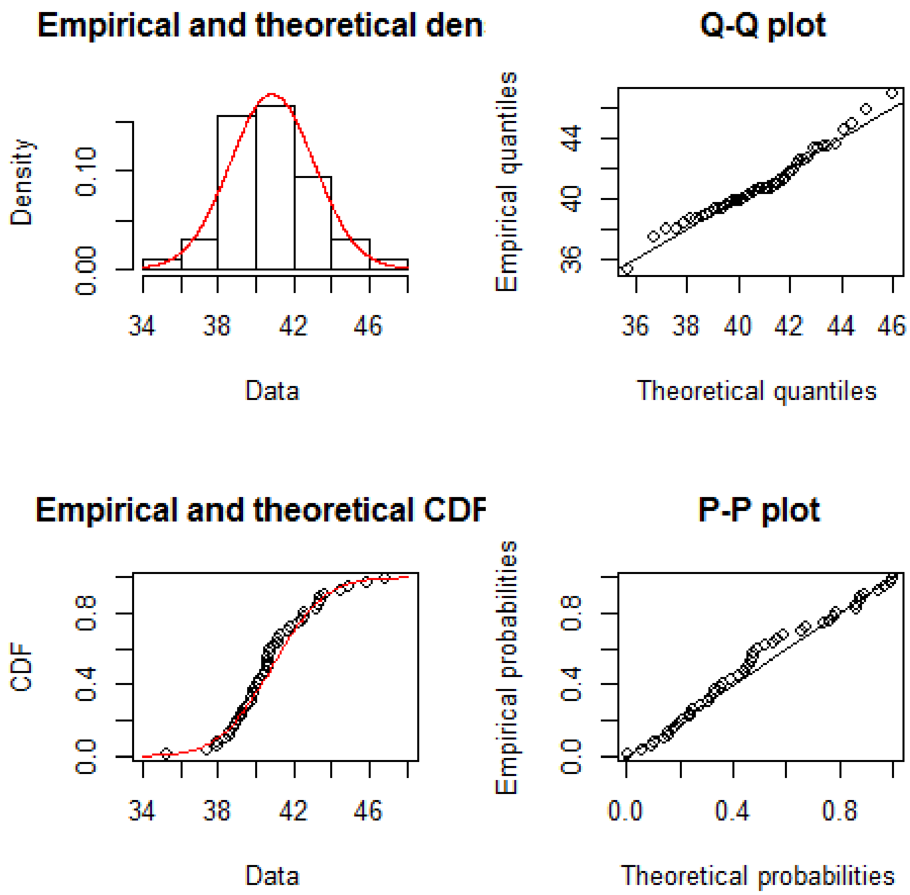



```
plot(fitdist(MDIndexPF66to13, "lnorm"))
```

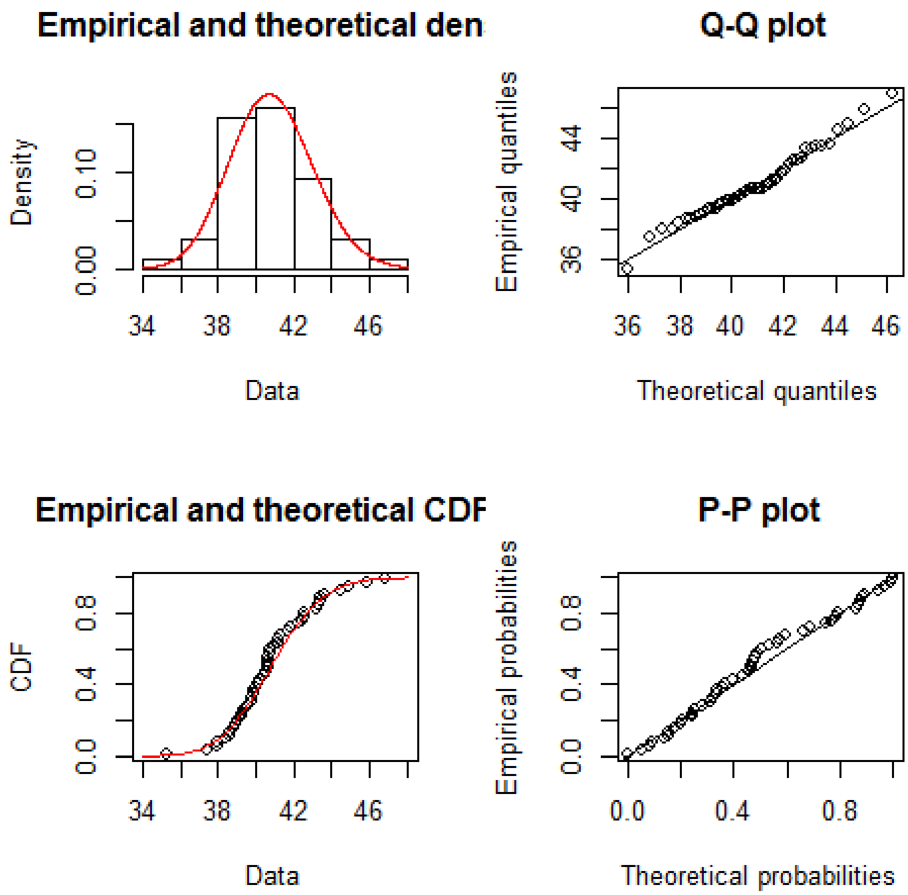


Northern Cardinal Trend Index Data

```
gofstat(list(fitdist(NCIndexPF66to13, "norm"), fitdist(NCIndexPF66to13,
        "lnorm")),fitnames = c("normal","lognormal"))

Goodness-of-fit statistics
                                 normal  lognormal
Kolmogorov-Smirnov statistic 0.11217744 0.10138763
Cramer-von Mises statistic   0.09178284 0.07016553
Anderson-Darling statistic   0.51098706 0.39811196

Goodness-of-fit criteria
                                 normal lognormal
Aikake's Information Criterion 217.2321  216.3121
Bayesian Information Criterion 220.9745  220.0545

plot(fitdist(NCIndexPF66to13, "norm"))
```




```
plot(fitdist(NCIndexPF66to13, "lnorm"))
```

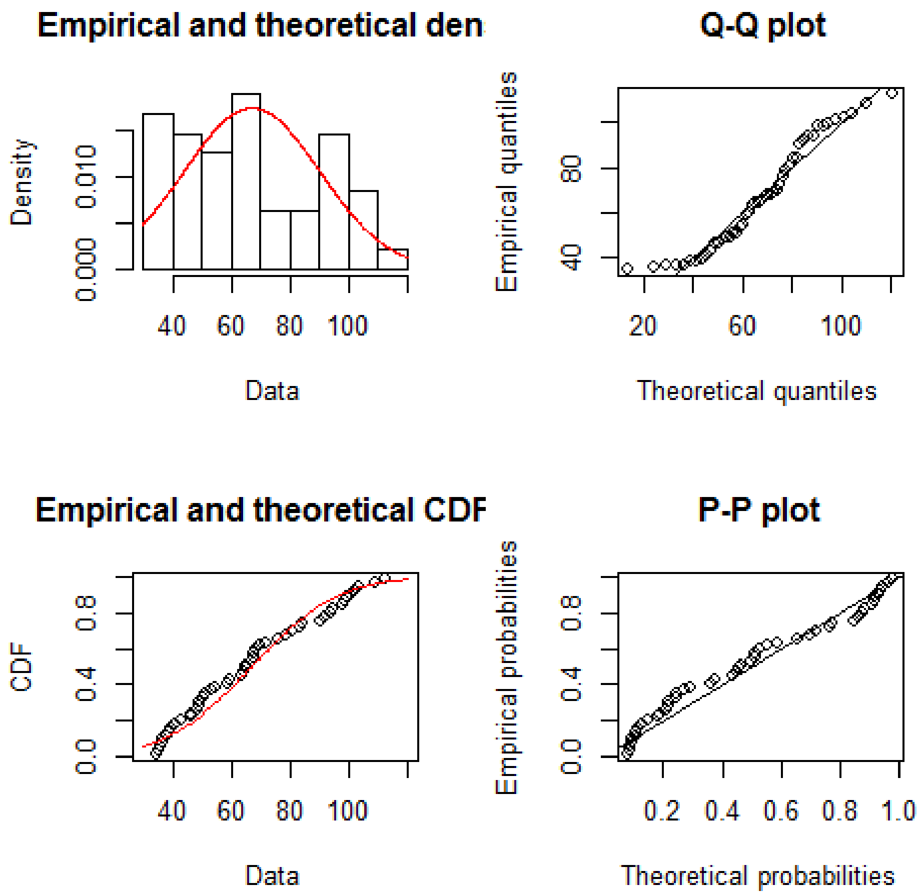


Northern Mockingbird Trend Index Data

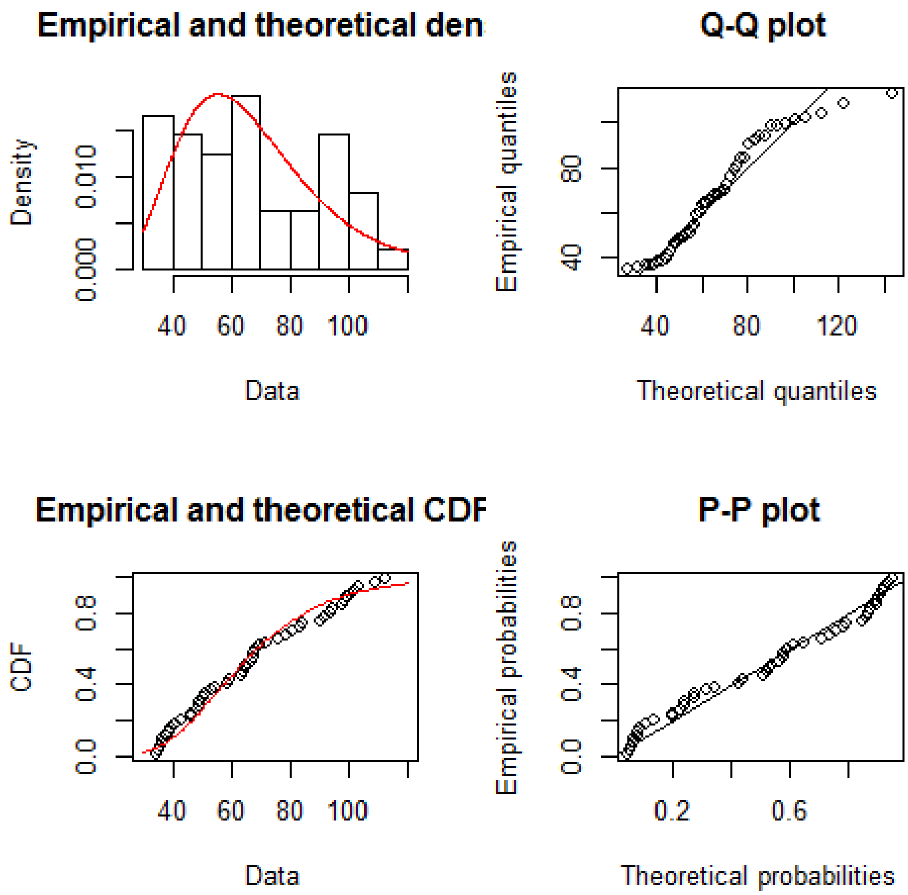
```
gofstat(list(fitdist(NMIndexPF66to13, "norm"), fitdist(NMIndexPF66to13,
        "lnorm")),fitnames = c("normal","lognormal"))
```

```
Goodness-of-fit statistics
                                normal   lognormal
Kolmogorov-Smirnov statistic 0.1100049 0.09510032
Cramer-von Mises statistic   0.1310815 0.09637673
Anderson-Darling statistic   0.9297053 0.74769324

Goodness-of-fit criteria
                                 normal lognormal
Aikake's Information Criterion 441.6778  438.6402
Bayesian Information Criterion 445.4202  442.3826
```

```
plot(fitdist(NMIndexPF66to13, "norm"))
```



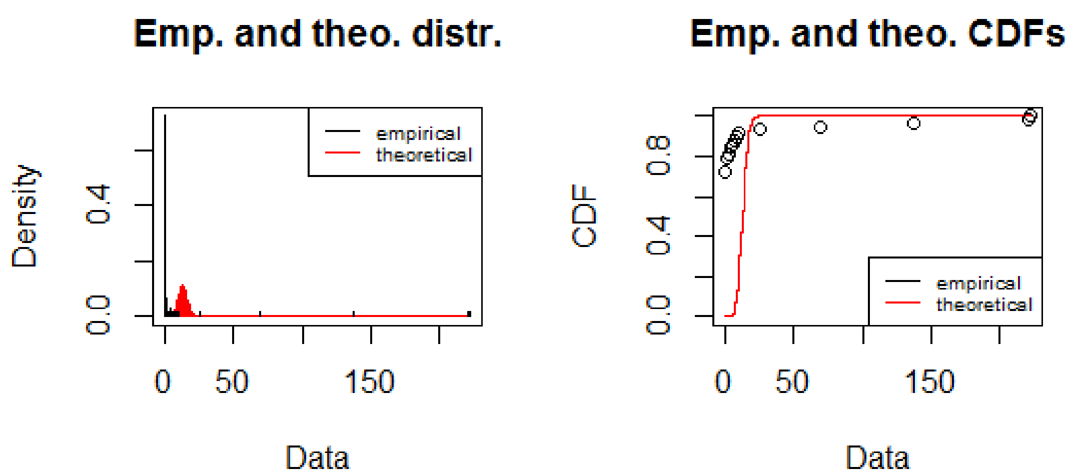

```
plot(fitdist(NMIndexPF66to13, "lnorm"))
```



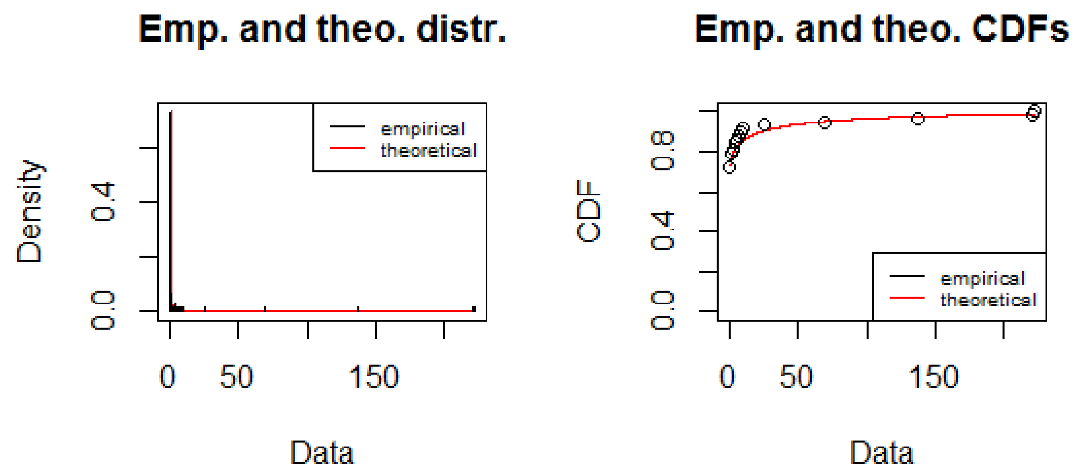
Transmission Count Data

Human SLE Count Data

```
plot(fitdist(HumSLEPF58to15, "pois"))
```



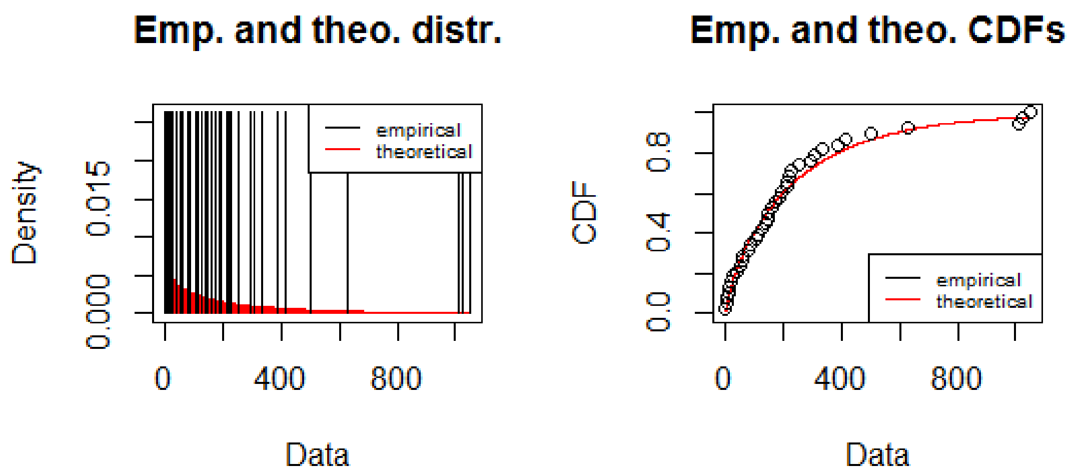

```
plot(fitdist(HumSLEPF58to15, "nbinom"))
```



Sentinel Chicken Seroconversions Flavivirus Count Data

```
gofstat(list(fitdist(SCFlaviPF78to15, "pois"), fitdist(SCFlaviPF78to15,
        "nbinom")),fitnames = c("Poisson","negative binomial"))

Chi-squared statistic:  7.829065e+84 3.088814
Degree of freedom of the Chi-squared distribution:  5 4
Chi-squared p-value:  0 0.5430742
   the p-value may be wrong with some theoretical counts < 5
Chi-squared table:
       obscounts theo Poisson theo negative binomial
<= 10          5 3.193229e-84               3.466152
<= 54          5 1.437744e-44               7.179539
<= 109         5 9.339203e-19               5.843009
<= 155         5 5.576040e-07               3.645904
<= 213         5 2.834518e+00               3.579746
<= 308         5 3.516538e+01               4.230030
> 308          8 1.008057e-04              10.055620

Goodness-of-fit criteria
                                Poisson negative binomial
Aikake's Information Criterion 9864.054          492.5814
Bayesian Information Criterion 9865.692          495.8566

plot(fitdist(SCFlaviPF78to15, "pois"))
```


```
plot(fitdist(SCFlaviPF78to15, "nbinom"))
```

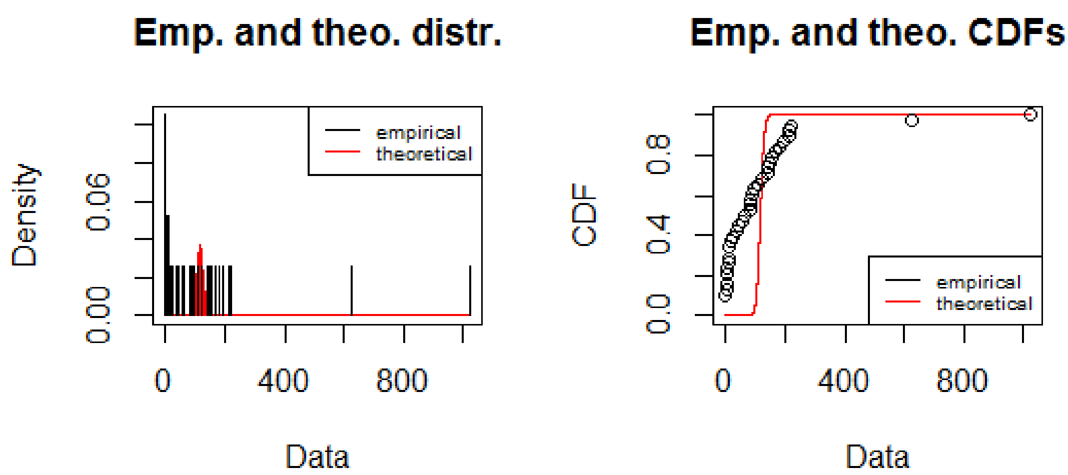


Sentinel Chicken Seroconversions SLE Count Data

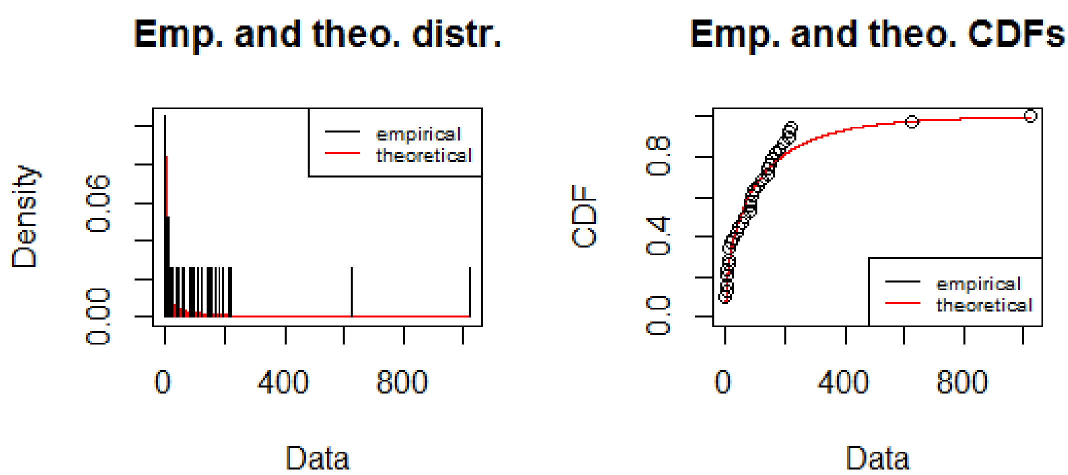
```
gofstat(list(fitdist(SCSLEPF78to15, "pois"), fitdist(SCSLEPF78to15,
        "nbinom")),fitnames = c("Poisson","negative binomial"))

Chi-squared statistic:  1.141973e+46 6.818649
Degree of freedom of the Chi-squared distribution:  5 4
Chi-squared p-value:  0 0.1457878
   the p-value may be wrong with some theoretical counts < 5
Chi-squared table:
       obscounts theo Poisson theo negative binomial
<= 2           5 2.189194e-45              5.626289
<= 9           5 3.479410e-36              4.194254
<= 21          5 1.533796e-25              4.032481
<= 80          5 1.228404e-02              9.414020
<= 105         5 6.887959e+00              2.277792
<= 155         5 3.109294e+01              3.288626
> 155          8 6.818445e-03              9.166538

Goodness-of-fit criteria
                                Poisson negative binomial
```

```
Aikake's Information Criterion 7314.927          422.4761
Bayesian Information Criterion 7316.564          425.7512
```

```r
plot(fitdist(SCSLEPF78to15, "pois"))
```




```r
plot(fitdist(SCSLEPF78to15, "nbinom"))
```



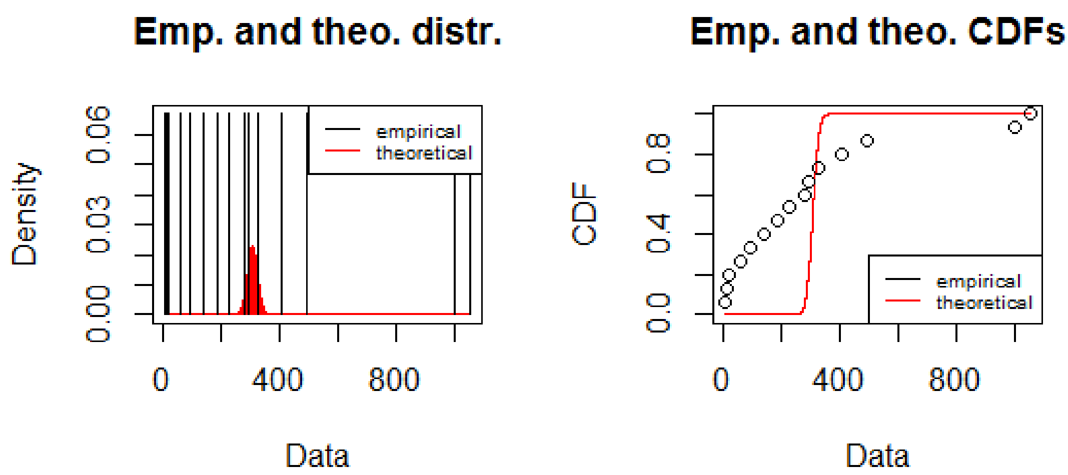
Sentinel Chicken Seroconversions West Nile Count Data

```r
gofstat(list(fitdist(SCWNPF01to15, "pois"), fitdist(SCWNPF01to15, "nbinom")),
        fitnames = c("Poisson","negative binomial"))
```

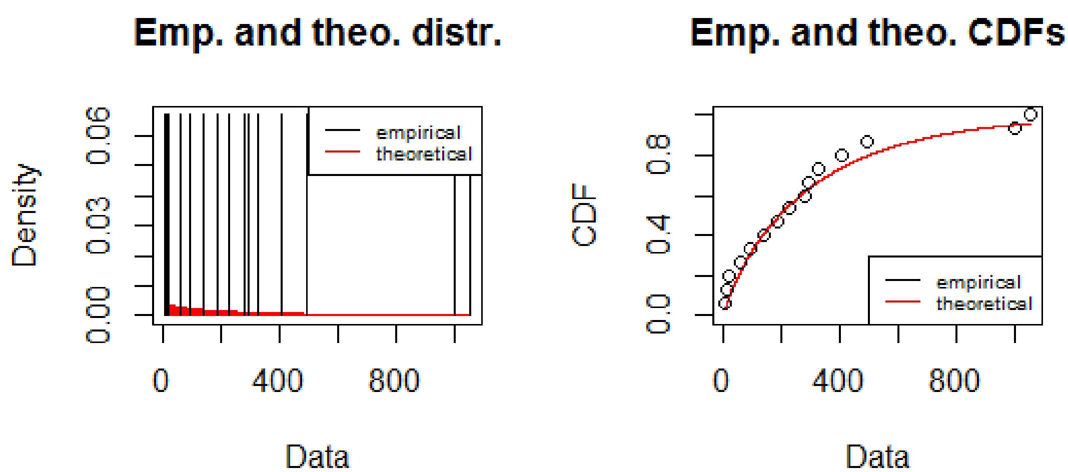
```
Chi-squared statistic:  1.135931e+103 3.461011
Degree of freedom of the Chi-squared distribution:  3 2
Chi-squared p-value:  0 0.1771949
   the p-value may be wrong with some theoretical counts < 5
Chi-squared table:
       obscounts  theo Poisson theo negative binomial
<= 18          3 7.923018e-103             1.421954
<= 135         3  1.160337e-26             4.632383
<= 278         3  1.008676e+00             3.219105
<= 401         3  1.399132e+01             1.755545
> 401          3  8.124348e-07             3.971014
```

```
Goodness-of-fit criteria
                               Poisson negative binomial
Aikake's Information Criterion 4555.084          205.0827
Bayesian Information Criterion 4555.792          206.4988
```

```
plot(fitdist(SCWNPF01to15, "pois"))
```




```
plot(fitdist(SCWNPF01to15, "nbinom"))
```



Transformations

Data transformation needed to meet normal distribution assumption for ARIMA modeling.

BBS Data Transformation

BBs Counts data are not going to be used in analysis, so no transformation needed.

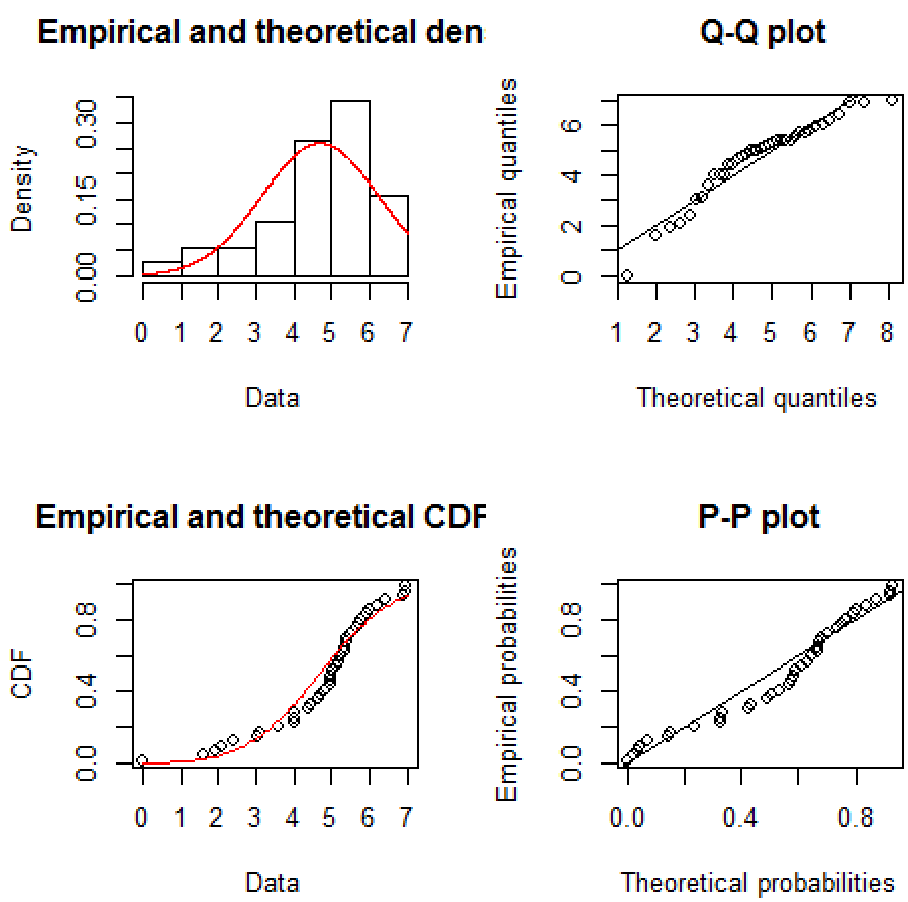
BBS Averages data fit a normal distribution, so no transformation needed.

Transmission Data Transformation

Transmission data log transformed to fit a normal distribution model.

Flavivirus Sentinel Chicken Seroconversion Count Data

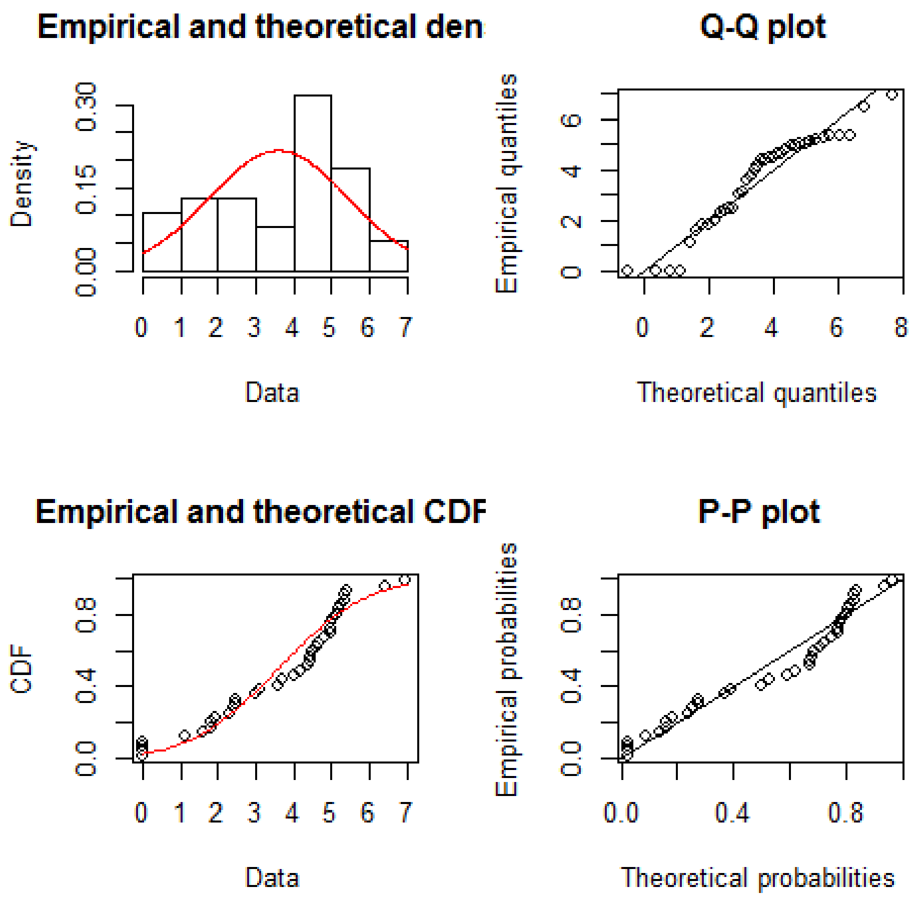
```
gofstat(fitdist(log(SCFlaviPF78to15+1), "norm"),fitnames = "normal")

Goodness-of-fit statistics
                                normal
Kolmogorov-Smirnov statistic 0.1518629
Cramer-von Mises statistic   0.1831915
Anderson-Darling statistic   1.0037106

Goodness-of-fit criteria
                                 normal
Aikake's Information Criterion 144.6611
Bayesian Information Criterion 147.9362

plot(fitdist(log(SCFlaviPF78to15+1), "norm"))
```



SLE Sentinel Chicken Seroconversion SLE Count Data

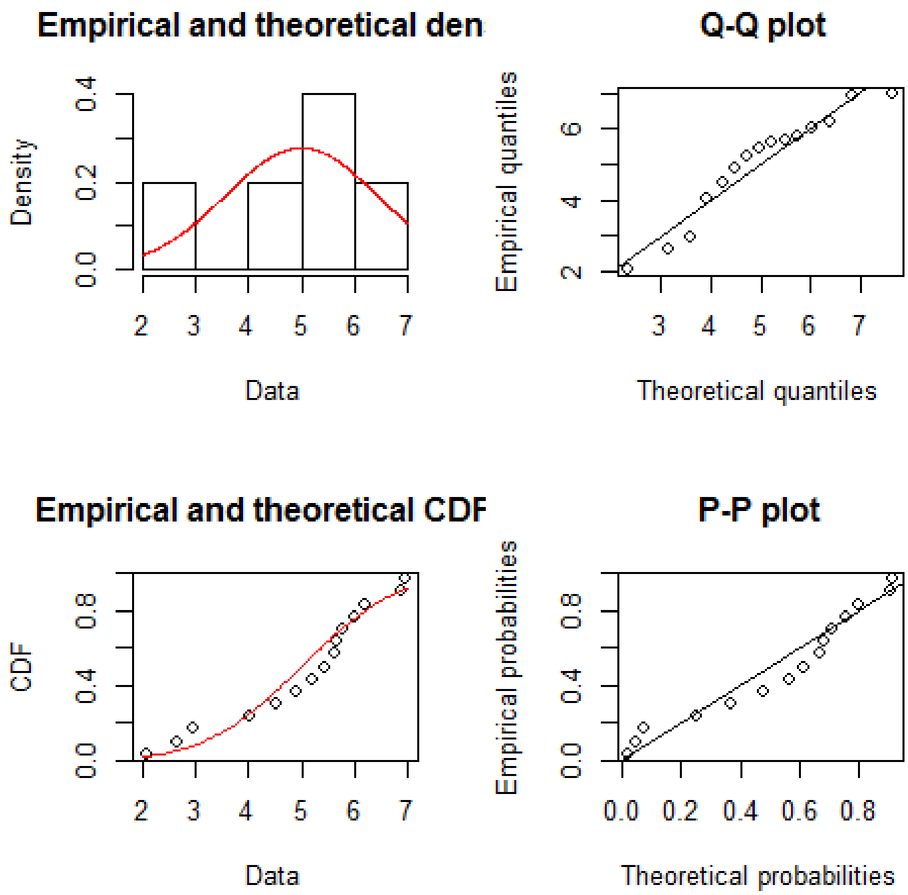
```
gofstat(fitdist(log(SCSLEPF78to15+1), "norm"),fitnames = "normal")

Goodness-of-fit statistics
                                normal
Kolmogorov-Smirnov statistic 0.1692318
Cramer-von Mises statistic   0.1826596
Anderson-Darling statistic   1.0691424

Goodness-of-fit criteria
                                 normal
Aikake's Information Criterion 158.1715
Bayesian Information Criterion 161.4467
```

```
plot(fitdist(log(SCSLEPF78to15+1), "norm"))
```



WN Sentinel Chicken Seroconversion Count Data

```
gofstat(fitdist(log(SCWNPF01to15+1), "norm"),fitnames = "normal")

Goodness-of-fit statistics
                                 normal
Kolmogorov-Smirnov statistic 0.16361837
Cramer-von Mises statistic   0.08140702
Anderson-Darling statistic   0.49611714

Goodness-of-fit criteria
                                 normal
Aikake's Information Criterion 57.52724
Bayesian Information Criterion 58.94334

plot(fitdist(log(SCWNPF01to15+1), "norm"))
```



Check for Stationarity

This is just a preliminary check for diffrences needed to meeet stationary assumption for ARIMA modeling. Fitting of ARIMA models in Model Fitting phase will check for and apply differencing as needed.

The Augmented Dickey-Fuller (ADF) t-statistic test: small p-values suggest the data is stationary and doesn't need to be differenced stationarity.
The Kwiatkowski-Phillips-Schmidt-Shin (KPSS) test; here accepting the null hypothesis means that the series is stationarity, and small p-values suggest that the series is not stationary and a differencing is required.

BBS Data Differences

Blue Jay Averages

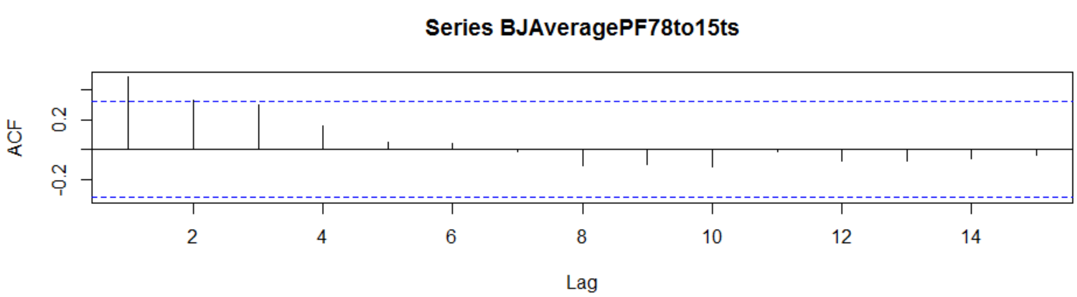
```
cat("Calculated differences needed for stationary via KPSS: ",
    ndiffs(BJAveragePF78to15ts, alpha=0.05, test="kpss"))

Calculated differences needed for stationary via KPSS:  1

cat("\n")

adf.test(BJAveragePF78to15ts, alternative = "stationary")


    Augmented Dickey-Fuller Test

data:  BJAveragePF78to15ts
Dickey-Fuller = -1.7219, Lag order = 3, p-value = 0.682
alternative hypothesis: stationary
```

```
adf.test(diff(BJAveragePF78to15ts), alternative = "stationary")


	Augmented Dickey-Fuller Test

data:  diff(BJAveragePF78to15ts)
Dickey-Fuller = -3.745, Lag order = 3, p-value = 0.03559
alternative hypothesis: stationary
```

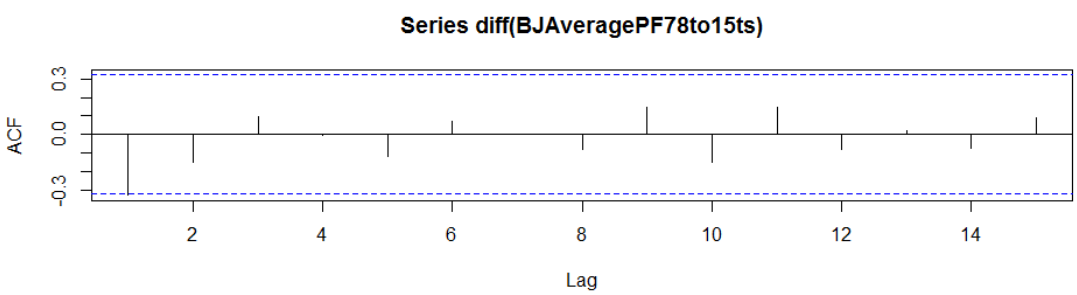
```
kpss.test(BJAveragePF78to15ts)


	KPSS Test for Level Stationarity

data:  BJAveragePF78to15ts
KPSS Level = 0.48798, Truncation lag parameter = 1, p-value = 0.04437
```

```
kpss.test(diff(BJAveragePF78to15ts))


	KPSS Test for Level Stationarity

data:  diff(BJAveragePF78to15ts)
KPSS Level = 0.079917, Truncation lag parameter = 1, p-value = 0.1
```

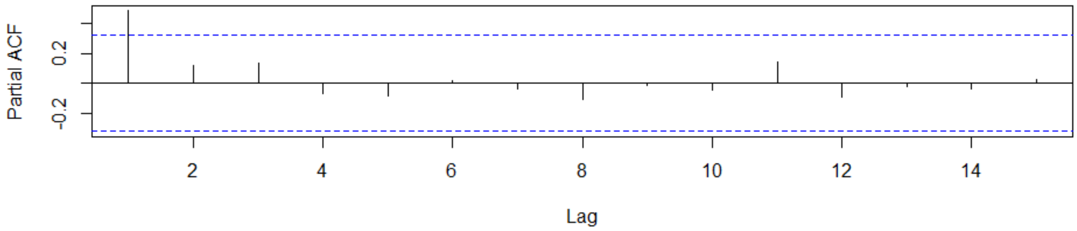
```
acf(BJAveragePF78to15ts)
```



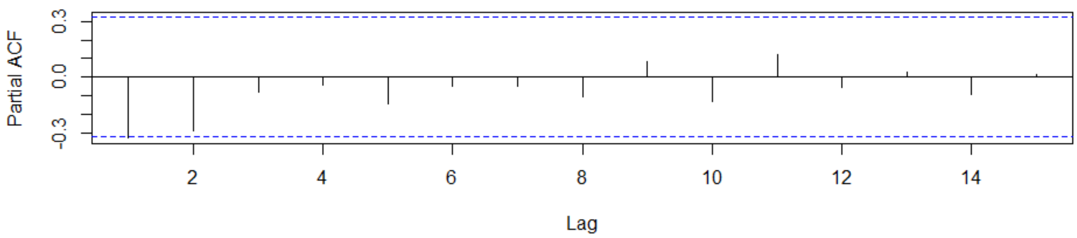

```
acf(diff(BJAveragePF78to15ts))
```




```
pacf(BJAveragePF78to15ts, main="")
```




```
pacf(diff(BJAveragePF78to15ts), main="")
```



Common Grackle Averages

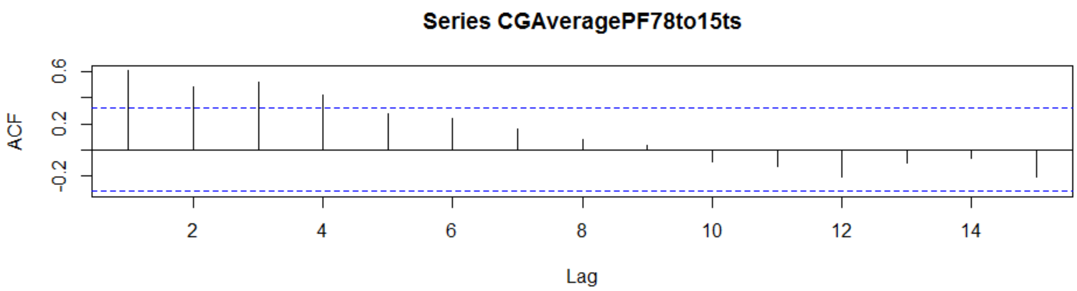
```
cat("Calculated differences needed for stationary via KPSS: ",
    ndiffs(CGAveragePF78to15ts, alpha=0.05, test="kpss"))

Calculated differences needed for stationary via KPSS:  1

cat("\n")

Box.test(CGAveragePF78to15ts, type = "Ljung-Box")


    Box-Ljung test

data:  CGAveragePF78to15ts
X-squared = 14.99, df = 1, p-value = 0.0001081

adf.test(CGAveragePF78to15ts, alternative = "stationary")


    Augmented Dickey-Fuller Test

data:  CGAveragePF78to15ts
Dickey-Fuller = -2.1757, Lag order = 3, p-value = 0.5047
alternative hypothesis: stationary

adf.test(diff(CGAveragePF78to15ts), alternative = "stationary")


    Augmented Dickey-Fuller Test

data:  diff(CGAveragePF78to15ts)
Dickey-Fuller = -3.2048, Lag order = 3, p-value = 0.1034
alternative hypothesis: stationary

kpss.test(CGAveragePF78to15ts)


    KPSS Test for Level Stationarity
```

```
data:  CGAveragePF78to15ts
KPSS Level = 0.878, Truncation lag parameter = 1, p-value = 0.01
```

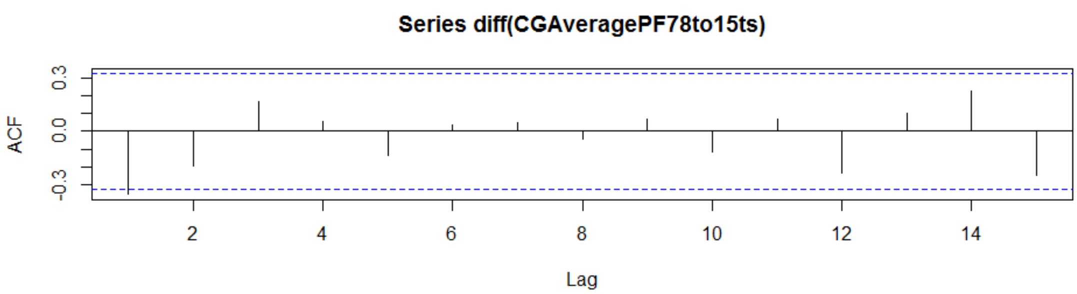
```r
kpss.test(diff(CGAveragePF78to15ts))
```

```

    KPSS Test for Level Stationarity

data:  diff(CGAveragePF78to15ts)
KPSS Level = 0.06529, Truncation lag parameter = 1, p-value = 0.1
```

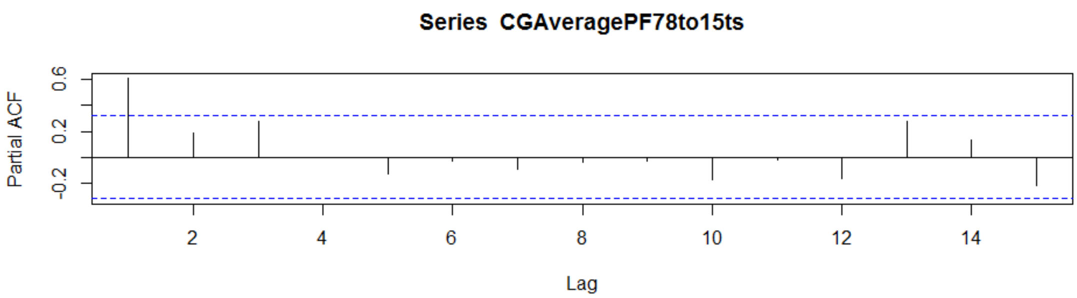
```r
acf(CGAveragePF78to15ts)
```



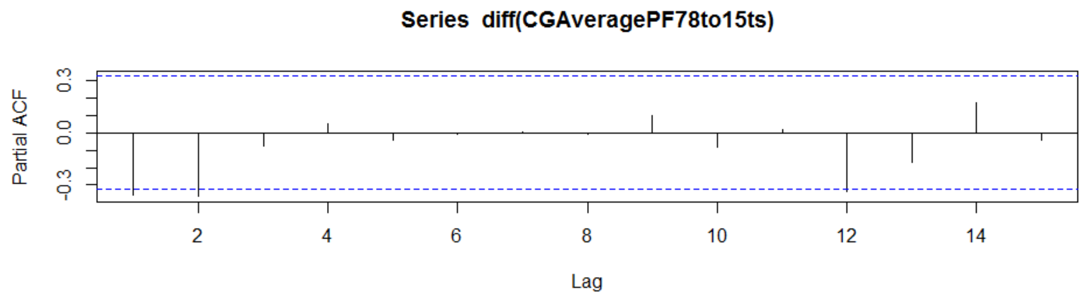

```r
acf(diff(CGAveragePF78to15ts))
```




```r
pacf(CGAveragePF78to15ts)
```

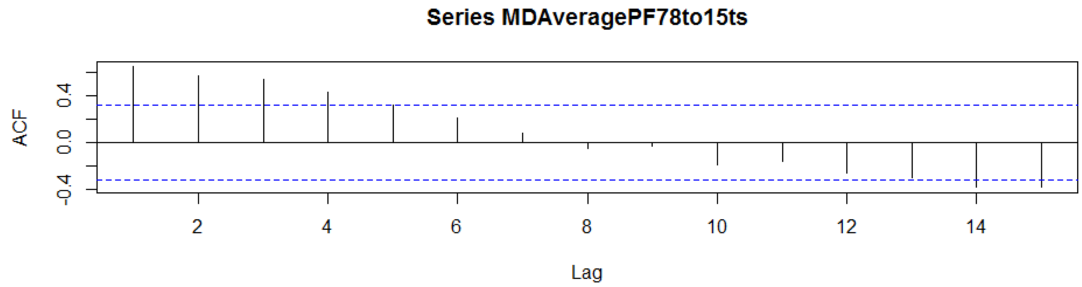



```r
pacf(diff(CGAveragePF78to15ts))
```

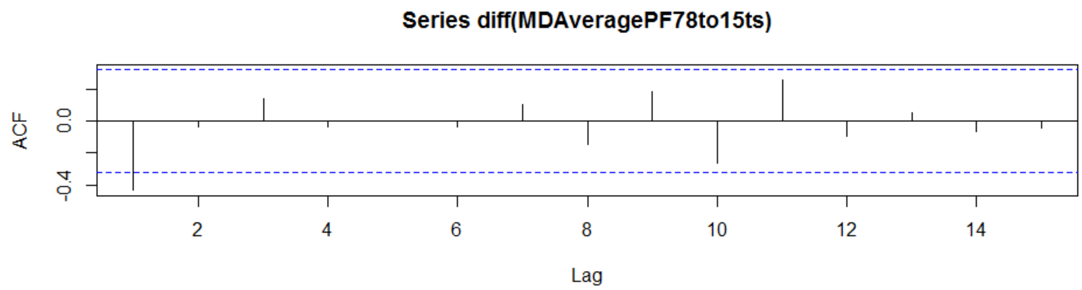


Mourning Dove Averages

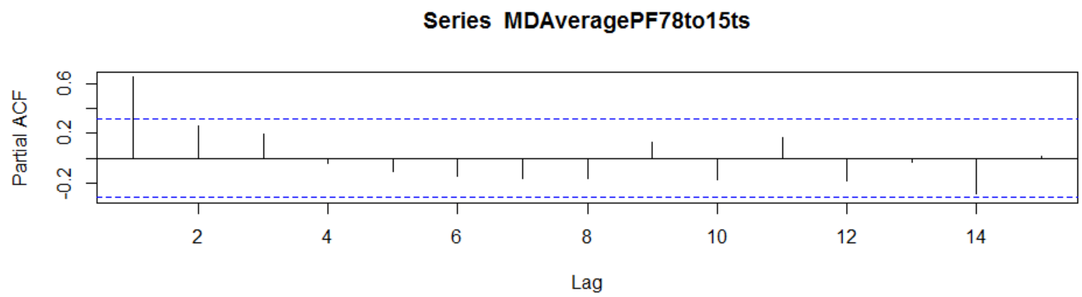
```
 cat("Calculated differences needed for stationary via KPSS: ",
     ndiffs(MDAveragePF78to15ts, alpha=0.05, test="kpss"))

Calculated differences needed for stationary via KPSS:  1

 cat("\n")

 Box.test(MDAveragePF78to15ts, type = "Ljung-Box")


    Box-Ljung test

data:  MDAveragePF78to15ts
X-squared = 17.297, df = 1, p-value = 3.197e-05

 adf.test(MDAveragePF78to15ts, alternative = "stationary")


    Augmented Dickey-Fuller Test

data:  MDAveragePF78to15ts
Dickey-Fuller = -1.0925, Lag order = 3, p-value = 0.9108
alternative hypothesis: stationary

 adf.test(diff(MDAveragePF78to15ts), alternative = "stationary")


    Augmented Dickey-Fuller Test

data:  diff(MDAveragePF78to15ts)
Dickey-Fuller = -3.477, Lag order = 3, p-value = 0.06154
alternative hypothesis: stationary

 kpss.test(MDAveragePF78to15ts)


    KPSS Test for Level Stationarity

data:  MDAveragePF78to15ts
KPSS Level = 0.5571, Truncation lag parameter = 1, p-value = 0.02881

 kpss.test(diff(MDAveragePF78to15ts))


    KPSS Test for Level Stationarity

data:  diff(MDAveragePF78to15ts)
KPSS Level = 0.15723, Truncation lag parameter = 1, p-value = 0.1

 acf(MDAveragePF78to15ts)
```



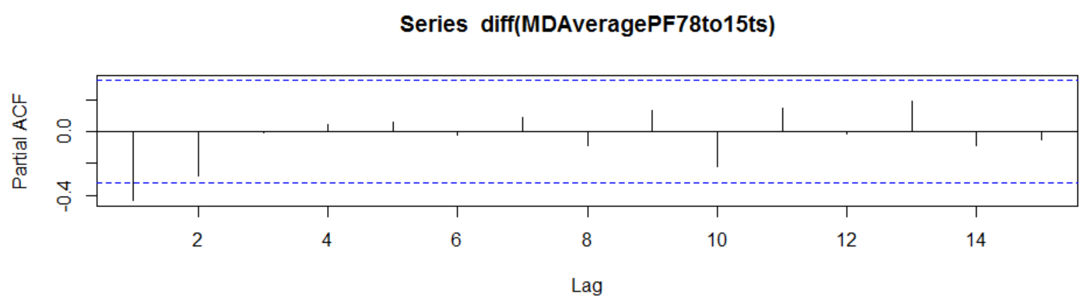

```r
acf(diff(MDAveragePF78to15ts))
```




```r
pacf(MDAveragePF78to15ts)
```




```r
pacf(diff(MDAveragePF78to15ts))
```



Northern Cardinal Averages

```r
cat("Calculated differences needed for stationary via KPSS: ",
    ndiffs(NCAveragePF78to15ts, alpha=0.05, test="kpss"))

Calculated differences needed for stationary via KPSS:  1

cat("\n")
```

```
Box.test(NCAveragePF78to15ts, type = "Ljung-Box")


    Box-Ljung test

data:  NCAveragePF78to15ts
X-squared = 33.028, df = 1, p-value = 9.084e-09
```

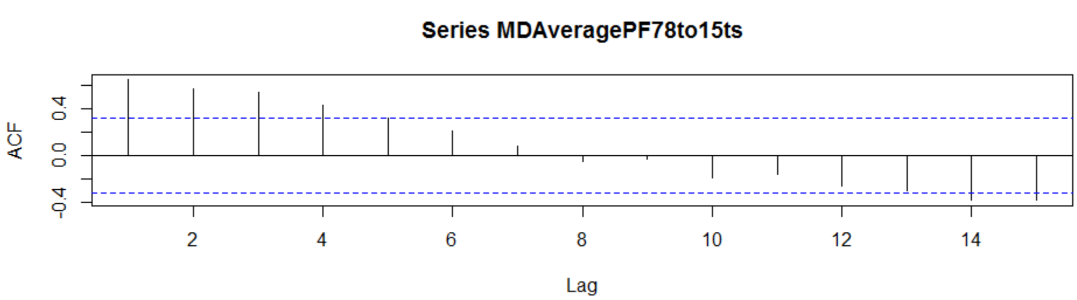
```
adf.test(NCAveragePF78to15ts, alternative = "stationary")


    Augmented Dickey-Fuller Test

data:  NCAveragePF78to15ts
Dickey-Fuller = -1.0784, Lag order = 3, p-value = 0.913
alternative hypothesis: stationary
```

```
adf.test(diff(NCAveragePF78to15ts), alternative = "stationary")


    Augmented Dickey-Fuller Test

data:  diff(NCAveragePF78to15ts)
Dickey-Fuller = -3.6576, Lag order = 3, p-value = 0.04226
alternative hypothesis: stationary
```

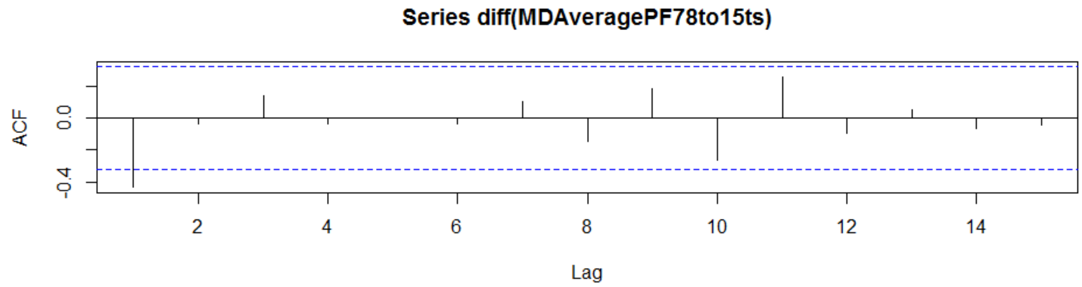
```
kpss.test(NCAveragePF78to15ts)


    KPSS Test for Level Stationarity

data:  NCAveragePF78to15ts
KPSS Level = 1.7308, Truncation lag parameter = 1, p-value = 0.01
```

```
kpss.test(diff(NCAveragePF78to15ts))


    KPSS Test for Level Stationarity

data:  diff(NCAveragePF78to15ts)
KPSS Level = 0.11765, Truncation lag parameter = 1, p-value = 0.1
```

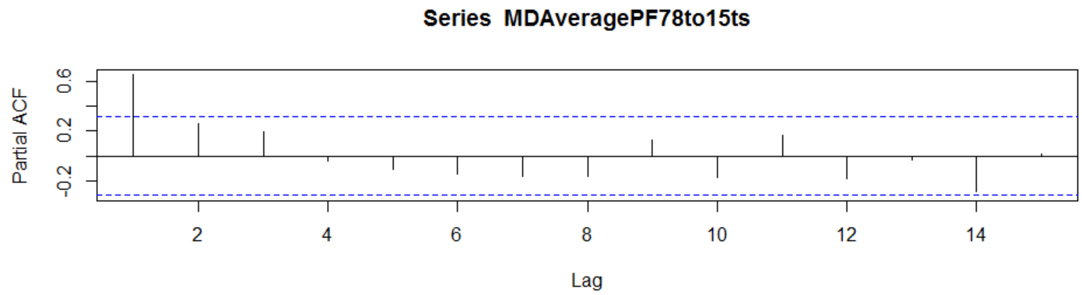
```
acf(MDAveragePF78to15ts)
```



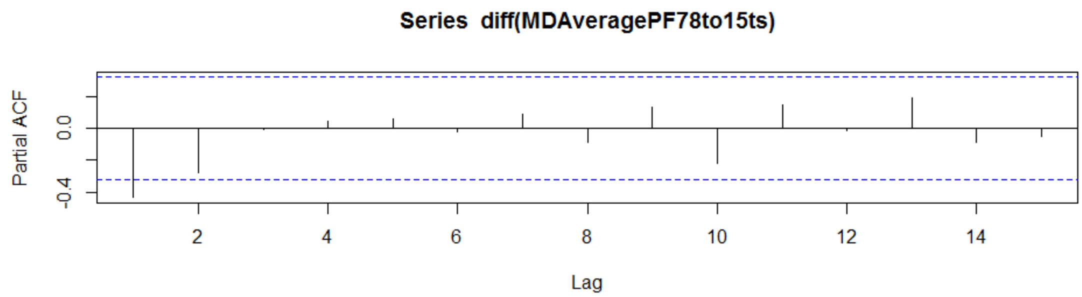

```
acf(diff(MDAveragePF78to15ts))
```




```r
pacf(MDAveragePF78to15ts)
```




```r
pacf(diff(MDAveragePF78to15ts))
```



Northern Mockingbird Averages

```r
cat("Calculated differences needed for stationary via KPSS: ",
    ndiffs(NMAveragePF78to15ts, alpha=0.05, test="kpss"))
```

```
Calculated differences needed for stationary via KPSS:  1
```

```r
cat("\n")
```

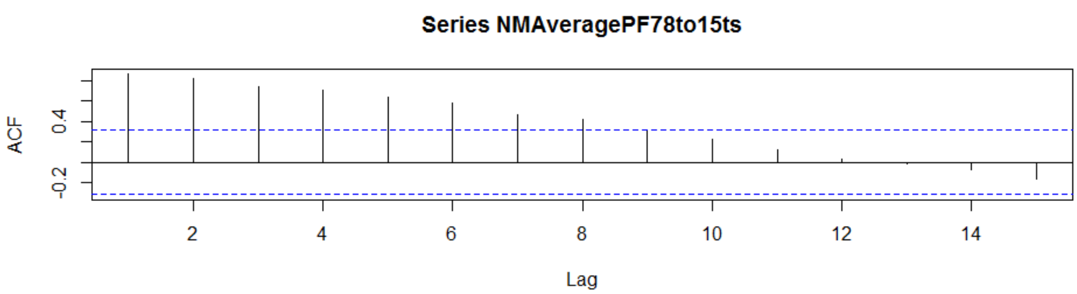
```r
Box.test(NMAveragePF78to15ts, type = "Ljung-Box")
```

```

    Box-Ljung test

data:  NMAveragePF78to15ts
X-squared = 30.5, df = 1, p-value = 3.338e-08
```

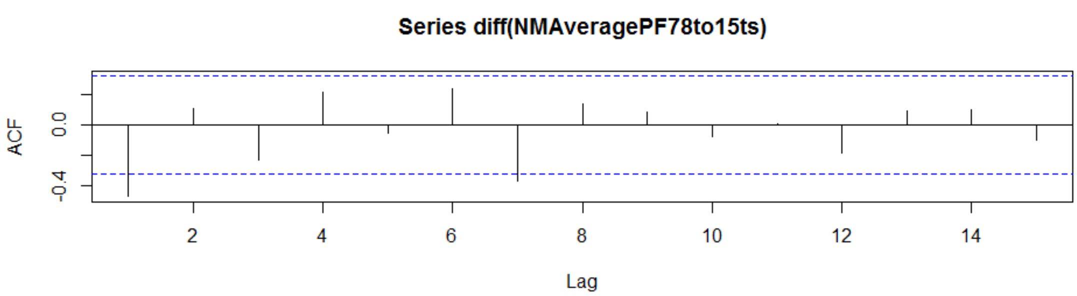
```r
adf.test(NMAveragePF78to15ts, alternative = "stationary")
```

```

    Augmented Dickey-Fuller Test

data:  NMAveragePF78to15ts
```

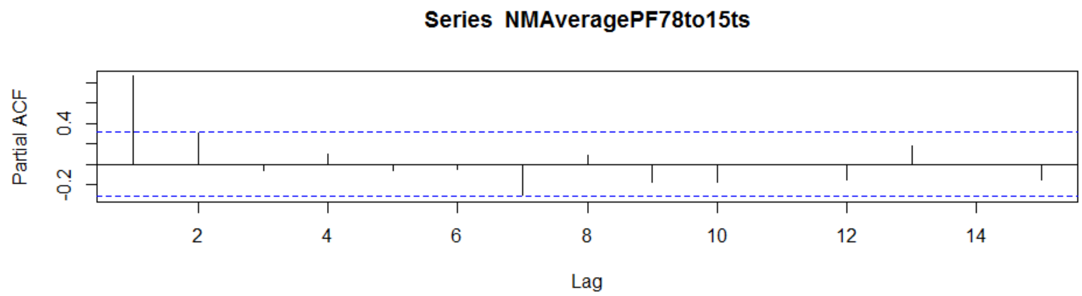
```
Dickey-Fuller = -1.4304, Lag order = 3, p-value = 0.7959
alternative hypothesis: stationary

 adf.test(diff(NMAveragePF78to15ts), alternative = "stationary")


    Augmented Dickey-Fuller Test

data:  diff(NMAveragePF78to15ts)
Dickey-Fuller = -3.9083, Lag order = 3, p-value = 0.02408
alternative hypothesis: stationary

 kpss.test(NMAveragePF78to15ts)


    KPSS Test for Level Stationarity

data:  NMAveragePF78to15ts
KPSS Level = 1.8095, Truncation lag parameter = 1, p-value = 0.01

 kpss.test(diff(NMAveragePF78to15ts))


    KPSS Test for Level Stationarity

data:  diff(NMAveragePF78to15ts)
KPSS Level = 0.062417, Truncation lag parameter = 1, p-value = 0.1

 acf(NMAveragePF78to15ts)
```



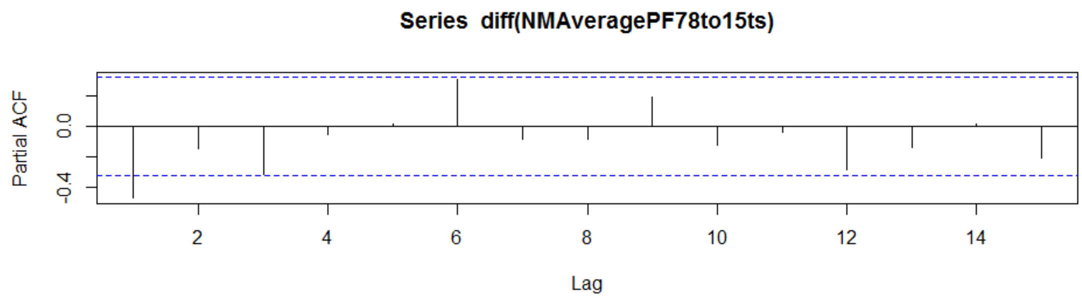

```
 acf(diff(NMAveragePF78to15ts))
```




```
 pacf(NMAveragePF78to15ts)
```




```
pacf(diff(NMAveragePF78to15ts))
```



Blue Jay Trend Index

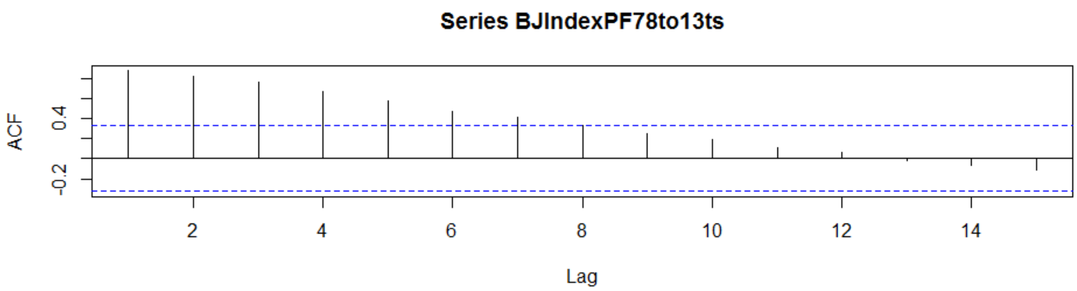
```
cat("Calculated differences needed for stationary via KPSS: ",
    ndiffs(BJIndexPF78to13ts, alpha=0.05, test="kpss"))

Calculated differences needed for stationary via KPSS:  1

cat("\n")

Box.test(BJIndexPF78to13ts, type = "Ljung-Box")


    Box-Ljung test

data:  BJIndexPF78to13ts
X-squared = 29.779, df = 1, p-value = 4.842e-08

adf.test(BJIndexPF78to13ts, alternative = "stationary")


    Augmented Dickey-Fuller Test

data:  BJIndexPF78to13ts
Dickey-Fuller = -1.9107, Lag order = 3, p-value = 0.6079
alternative hypothesis: stationary

adf.test(diff(BJIndexPF78to13ts), alternative = "stationary")


    Augmented Dickey-Fuller Test

data:  diff(BJIndexPF78to13ts)
Dickey-Fuller = -3.8106, Lag order = 3, p-value = 0.03143
alternative hypothesis: stationary

kpss.test(BJIndexPF78to13ts)


    KPSS Test for Level Stationarity
```

```
data:  BJIndexPF78to13ts
KPSS Level = 1.8122, Truncation lag parameter = 1, p-value = 0.01
```

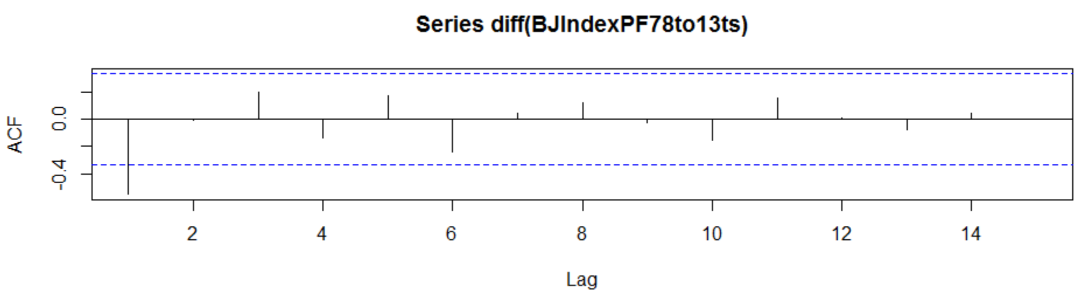
```
kpss.test(diff(BJIndexPF78to13ts))
```

```

    KPSS Test for Level Stationarity

data:  diff(BJIndexPF78to13ts)
KPSS Level = 0.055263, Truncation lag parameter = 1, p-value = 0.1
```

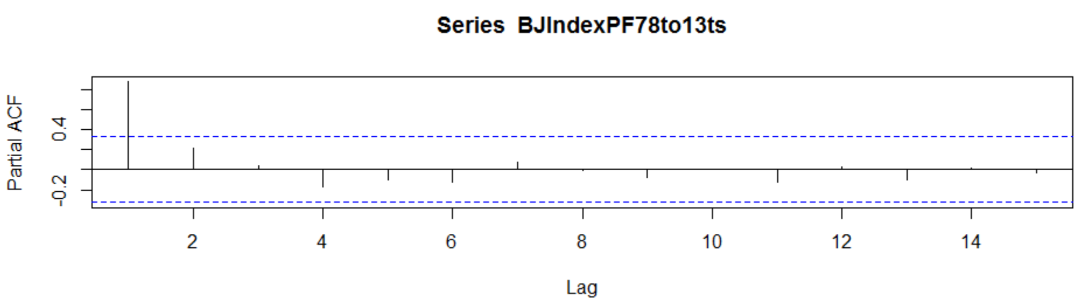
```
acf(BJIndexPF78to13ts)
```



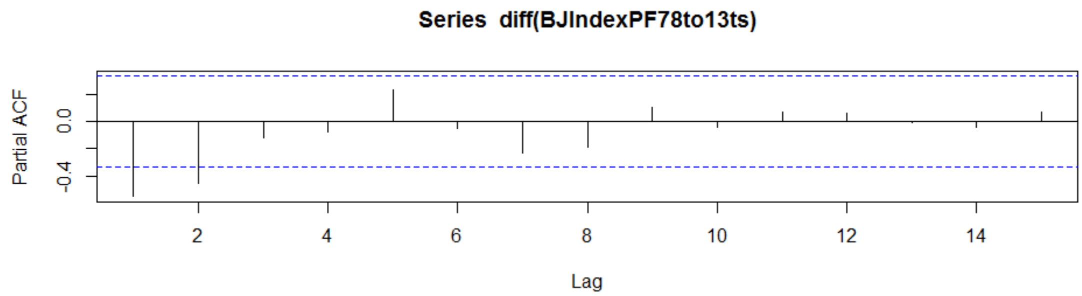

```
acf(diff(BJIndexPF78to13ts))
```




```
pacf(BJIndexPF78to13ts)
```

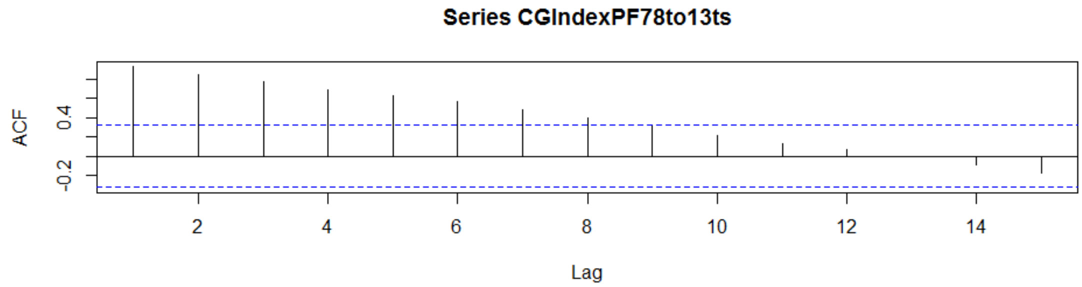



```
pacf(diff(BJIndexPF78to13ts))
```

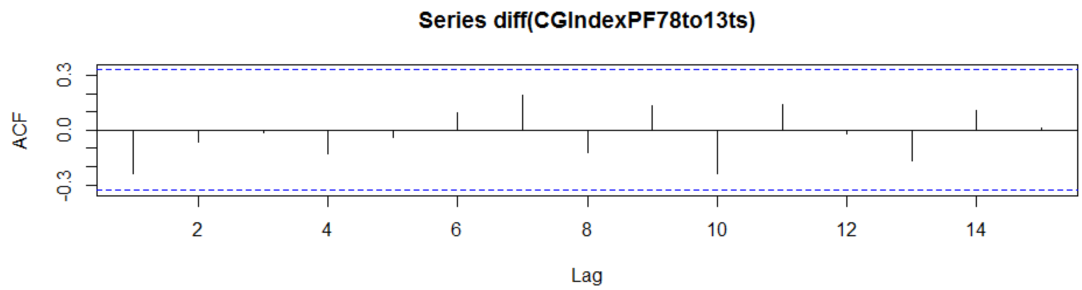


Common Grackle Trend Index

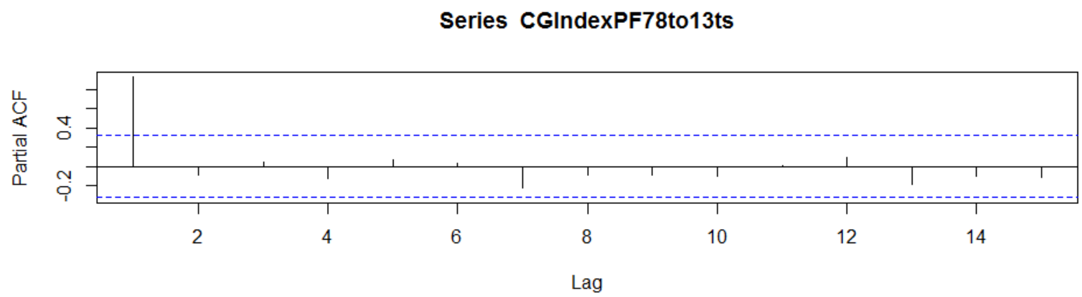
```
 cat("Calculated differences needed for stationary via KPSS: ",
     ndiffs(CGIndexPF78to13ts, alpha=0.05, test="kpss"))

Calculated differences needed for stationary via KPSS:  1

 cat("\n")

 Box.test(CGIndexPF78to13ts, type = "Ljung-Box")


    Box-Ljung test

data:  CGIndexPF78to13ts
X-squared = 33.512, df = 1, p-value = 7.083e-09

 adf.test(CGIndexPF78to13ts, alternative = "stationary")


    Augmented Dickey-Fuller Test

data:  CGIndexPF78to13ts
Dickey-Fuller = -1.9966, Lag order = 3, p-value = 0.5745
alternative hypothesis: stationary

 adf.test(diff(CGIndexPF78to13ts), alternative = "stationary")


    Augmented Dickey-Fuller Test

data:  diff(CGIndexPF78to13ts)
Dickey-Fuller = -3.6539, Lag order = 3, p-value = 0.04323
alternative hypothesis: stationary

 kpss.test(CGIndexPF78to13ts)


    KPSS Test for Level Stationarity

data:  CGIndexPF78to13ts
KPSS Level = 1.8122, Truncation lag parameter = 1, p-value = 0.01

 kpss.test(diff(CGIndexPF78to13ts))


    KPSS Test for Level Stationarity

data:  diff(CGIndexPF78to13ts)
KPSS Level = 0.080284, Truncation lag parameter = 1, p-value = 0.1

 acf(CGIndexPF78to13ts)
```



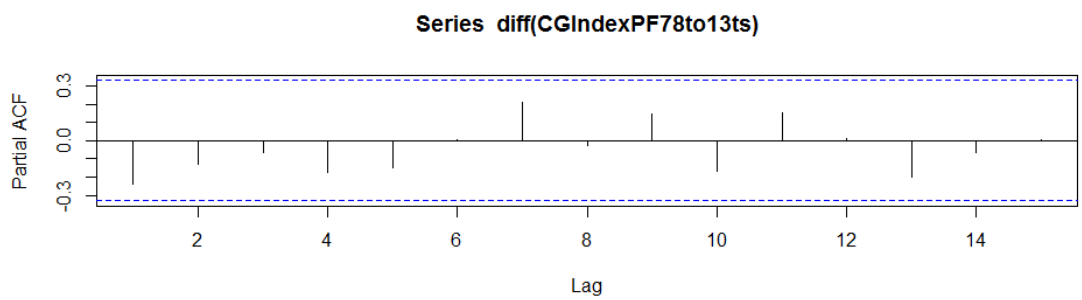

```r
acf(diff(CGIndexPF78to13ts))
```




```r
pacf(CGIndexPF78to13ts)
```




```r
pacf(diff(CGIndexPF78to13ts))
```



Mourning Dove Trend Index

```r
cat("Calculated differences needed for stationary via KPSS: ",
    ndiffs(MDIndexPF78to13ts, alpha=0.05, test="kpss"))

Calculated differences needed for stationary via KPSS:  1

cat("\n")
```

```
Box.test(MDIndexPF78to13ts, type = "Ljung-Box")


	Box-Ljung test

data:  MDIndexPF78to13ts
X-squared = 26.173, df = 1, p-value = 3.122e-07
```

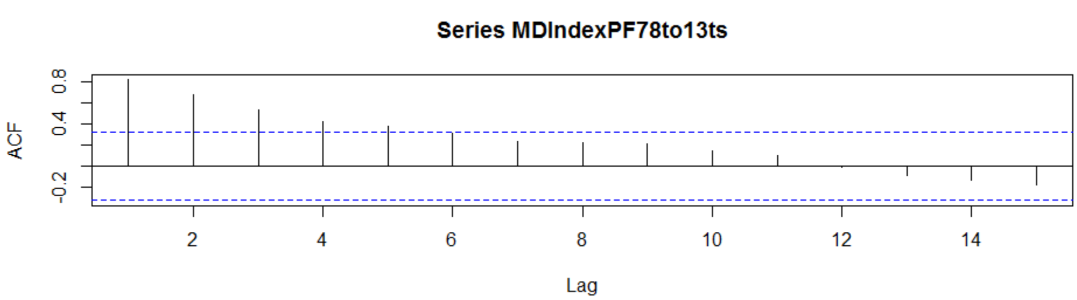
```
adf.test(MDIndexPF78to13ts, alternative = "stationary")


	Augmented Dickey-Fuller Test

data:  MDIndexPF78to13ts
Dickey-Fuller = -2.3474, Lag order = 3, p-value = 0.438
alternative hypothesis: stationary
```

```
adf.test(diff(MDIndexPF78to13ts), alternative = "stationary")


	Augmented Dickey-Fuller Test

data:  diff(MDIndexPF78to13ts)
Dickey-Fuller = -3.0753, Lag order = 3, p-value = 0.1556
alternative hypothesis: stationary
```

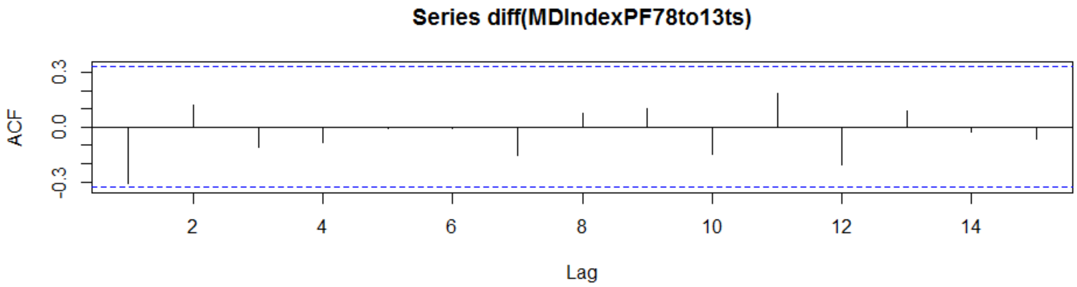
```
kpss.test(MDIndexPF78to13ts)


	KPSS Test for Level Stationarity

data:  MDIndexPF78to13ts
KPSS Level = 1.4472, Truncation lag parameter = 1, p-value = 0.01
```

```
kpss.test(diff(MDIndexPF78to13ts))


	KPSS Test for Level Stationarity

data:  diff(MDIndexPF78to13ts)
KPSS Level = 0.055945, Truncation lag parameter = 1, p-value = 0.1
```

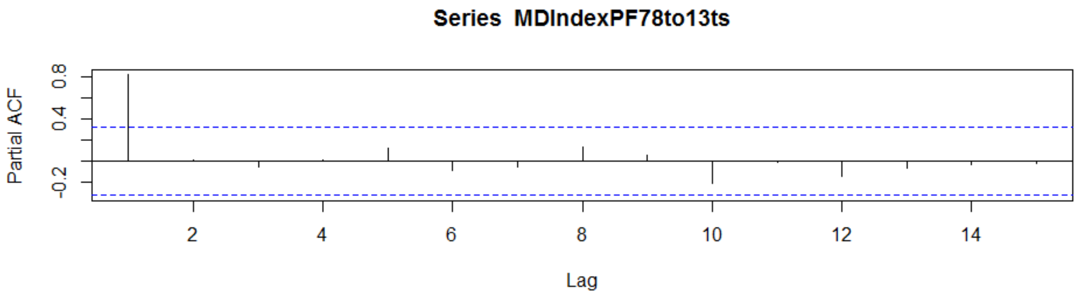
```
acf(MDIndexPF78to13ts)
```



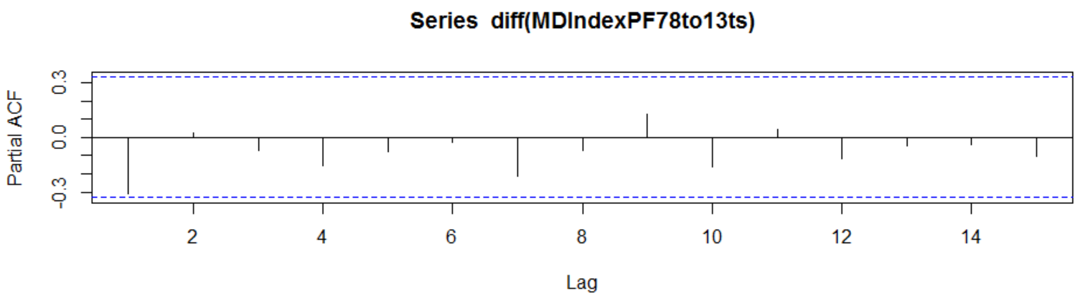

```
acf(diff(MDIndexPF78to13ts))
```




```
pacf(MDIndexPF78to13ts)
```



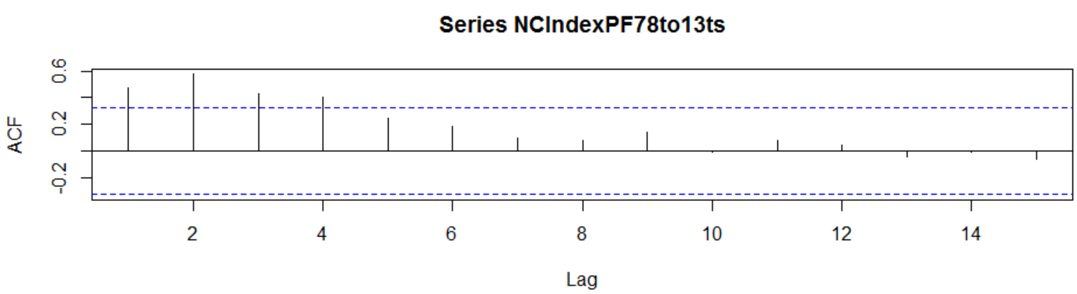

```
pacf(diff(MDIndexPF78to13ts))
```



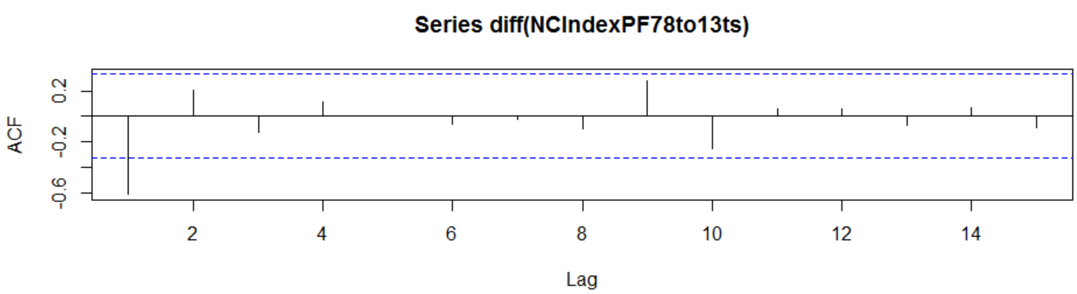
Northern Cardinal Trend Index

```
cat("Calculated differences needed for stationary via KPSS: ",
    ndiffs(NCIndexPF78to13ts, alpha=0.05, test="kpss"))

Calculated differences needed for stationary via KPSS:  1

cat("\n")

Box.test(NCIndexPF78to13ts, type = "Ljung-Box")


    Box-Ljung test

data:  NCIndexPF78to13ts
X-squared = 8.9004, df = 1, p-value = 0.002851

adf.test(NCIndexPF78to13ts, alternative = "stationary")


    Augmented Dickey-Fuller Test

data:  NCIndexPF78to13ts
```

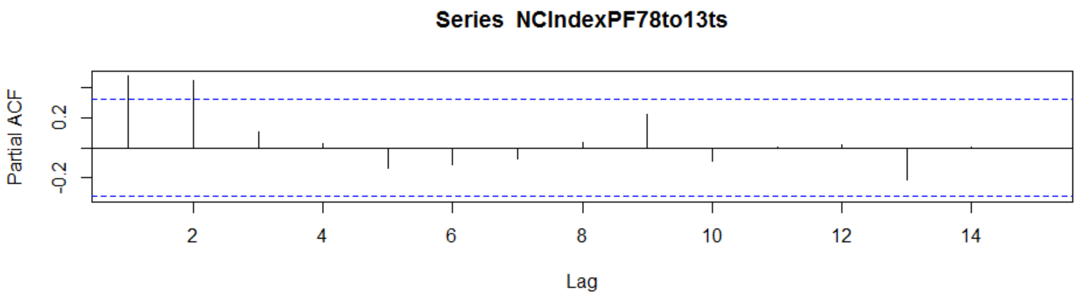
```
Dickey-Fuller = -2.5638, Lag order = 3, p-value = 0.3538
alternative hypothesis: stationary

 adf.test(diff(NCIndexPF78to13ts), alternative = "stationary")


    Augmented Dickey-Fuller Test

data:  diff(NCIndexPF78to13ts)
Dickey-Fuller = -4.017, Lag order = 3, p-value = 0.02047
alternative hypothesis: stationary

 kpss.test(NCIndexPF78to13ts)


    KPSS Test for Level Stationarity

data:  NCIndexPF78to13ts
KPSS Level = 1.3542, Truncation lag parameter = 1, p-value = 0.01

 kpss.test(diff(NCIndexPF78to13ts))


    KPSS Test for Level Stationarity

data:  diff(NCIndexPF78to13ts)
KPSS Level = 0.065347, Truncation lag parameter = 1, p-value = 0.1

 acf(NCIndexPF78to13ts)
```



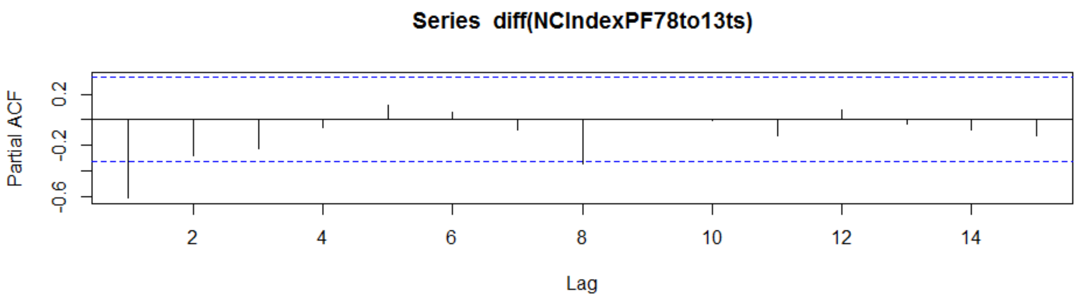

```
 acf(diff(NCIndexPF78to13ts))
```




```
 pacf(NCIndexPF78to13ts)
```



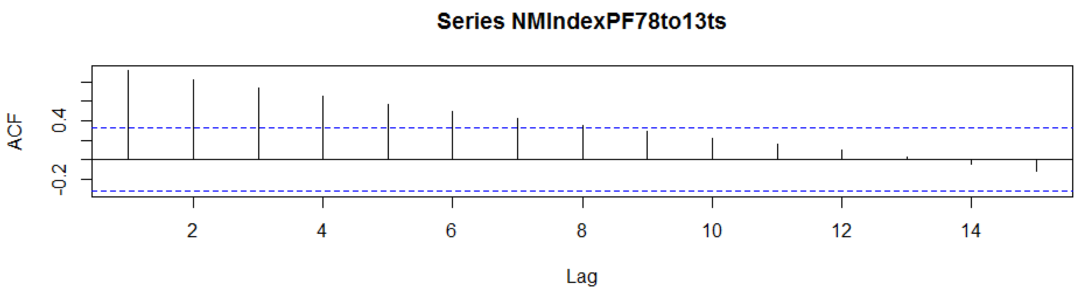

```
pacf(diff(NCIndexPF78to13ts))
```



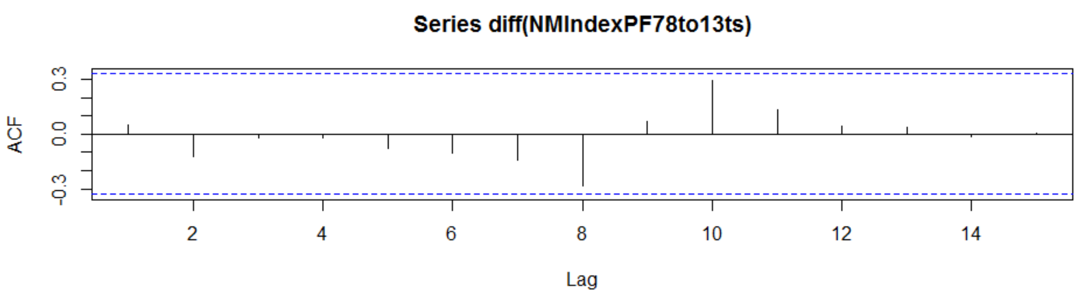
Northern Mockingbird Trend Index

```
cat("Calculated differences needed for stationary via KPSS: ",
    ndiffs(NMIndexPF78to13ts, alpha=0.05, test="kpss"))

Calculated differences needed for stationary via KPSS:  1

cat("\n")

Box.test(NMIndexPF78to13ts, type = "Ljung-Box")


    Box-Ljung test

data:  NMIndexPF78to13ts
X-squared = 32.588, df = 1, p-value = 1.139e-08

adf.test(NMIndexPF78to13ts, alternative = "stationary")


    Augmented Dickey-Fuller Test

data:  NMIndexPF78to13ts
Dickey-Fuller = -2.7806, Lag order = 3, p-value = 0.2694
alternative hypothesis: stationary

adf.test(diff(NMIndexPF78to13ts), alternative = "stationary")


    Augmented Dickey-Fuller Test

data:  diff(NMIndexPF78to13ts)
Dickey-Fuller = -3.0466, Lag order = 3, p-value = 0.1667
alternative hypothesis: stationary

kpss.test(NMIndexPF78to13ts)


    KPSS Test for Level Stationarity
```

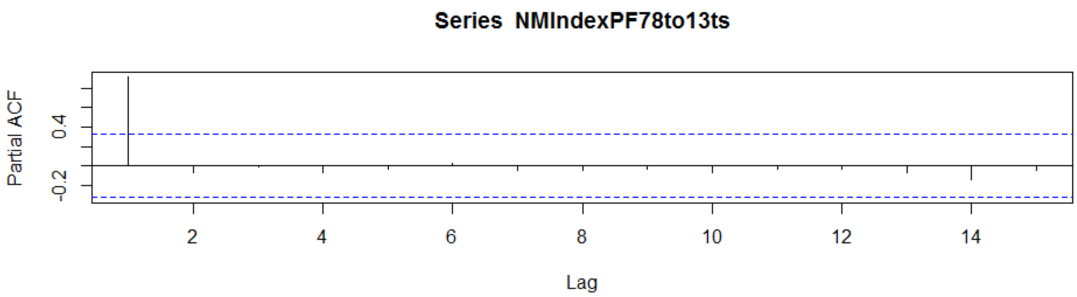
```
data:  NMIndexPF78to13ts
KPSS Level = 1.852, Truncation lag parameter = 1, p-value = 0.01

 kpss.test(diff(NMIndexPF78to13ts))


    KPSS Test for Level Stationarity

data:  diff(NMIndexPF78to13ts)
KPSS Level = 0.18858, Truncation lag parameter = 1, p-value = 0.1

 acf(NMIndexPF78to13ts)
```



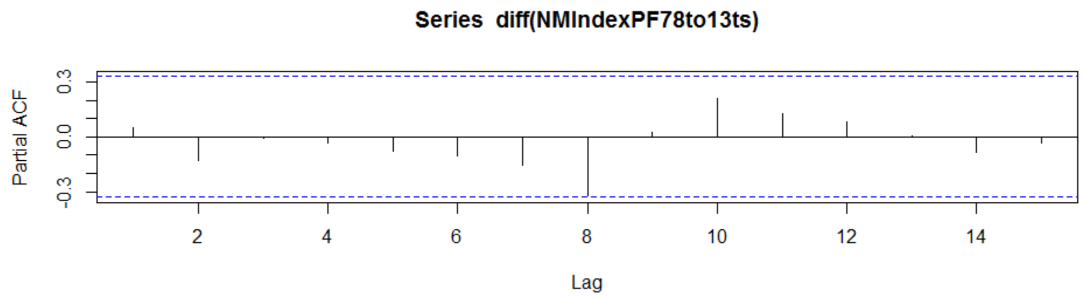

```
 acf(diff(NMIndexPF78to13ts))
```




```
 pacf(NMIndexPF78to13ts)
```




```
 pacf(diff(NMIndexPF78to13ts))
```

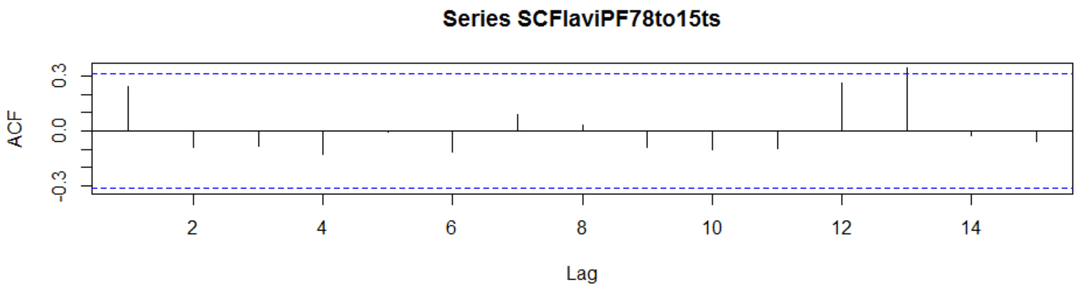


Transmission Data Differences

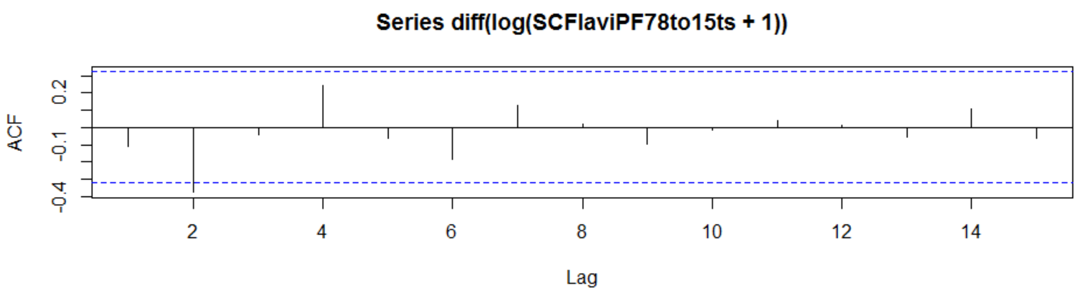
Flavivirus

```
 cat("Calculated differences needed for stationary via KPSS: ",
     ndiffs(log(SCFlaviPF78to15ts+1), alpha=0.05, test="kpss"))

Calculated differences needed for stationary via KPSS:  0

 cat("\n")

 Box.test(log(SCFlaviPF78to15ts+1), type = "Ljung-Box")


    Box-Ljung test

data:  log(SCFlaviPF78to15ts + 1)
X-squared = 2.29, df = 1, p-value = 0.1302

 adf.test(log(SCFlaviPF78to15ts+1), alternative = "stationary")


    Augmented Dickey-Fuller Test

data:  log(SCFlaviPF78to15ts + 1)
Dickey-Fuller = -2.9759, Lag order = 3, p-value = 0.1919
alternative hypothesis: stationary

 adf.test(diff(log(SCFlaviPF78to15ts+1)), alternative = "stationary")


    Augmented Dickey-Fuller Test

data:  diff(log(SCFlaviPF78to15ts + 1))
Dickey-Fuller = -4.0073, Lag order = 3, p-value = 0.02032
alternative hypothesis: stationary

 kpss.test(log(SCFlaviPF78to15ts+1))


    KPSS Test for Level Stationarity

data:  log(SCFlaviPF78to15ts + 1)
KPSS Level = 0.3425, Truncation lag parameter = 1, p-value = 0.1

 kpss.test(diff(log(SCFlaviPF78to15ts+1)))


    KPSS Test for Level Stationarity

data:  diff(log(SCFlaviPF78to15ts + 1))
KPSS Level = 0.06706, Truncation lag parameter = 1, p-value = 0.1
```

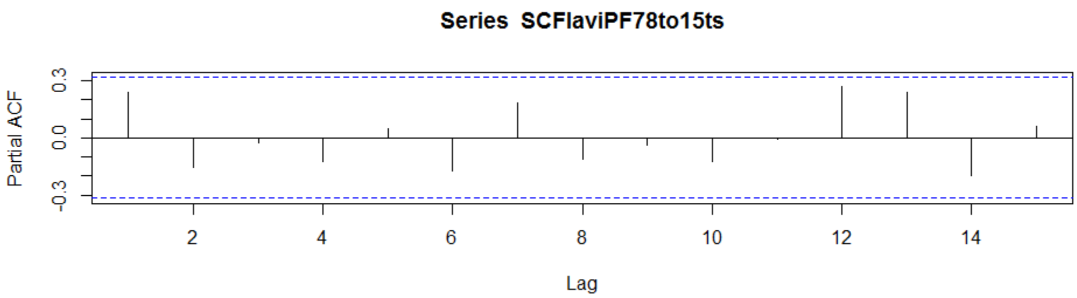
```r
acf(SCFlaviPF78to15ts)
```



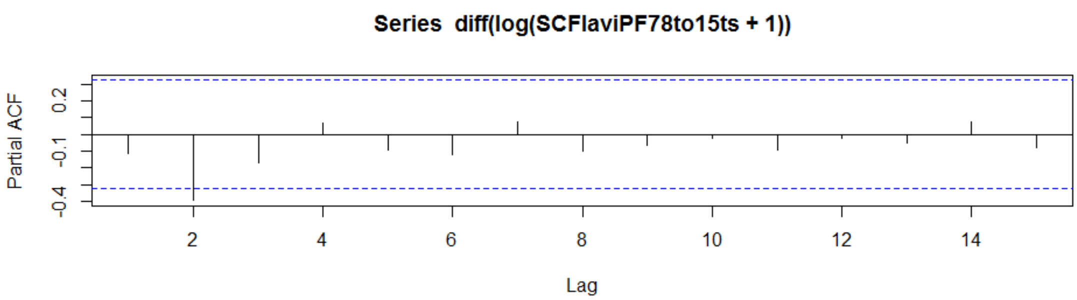

```r
acf(diff(log(SCFlaviPF78to15ts+1)))
```




```r
pacf(SCFlaviPF78to15ts)
```



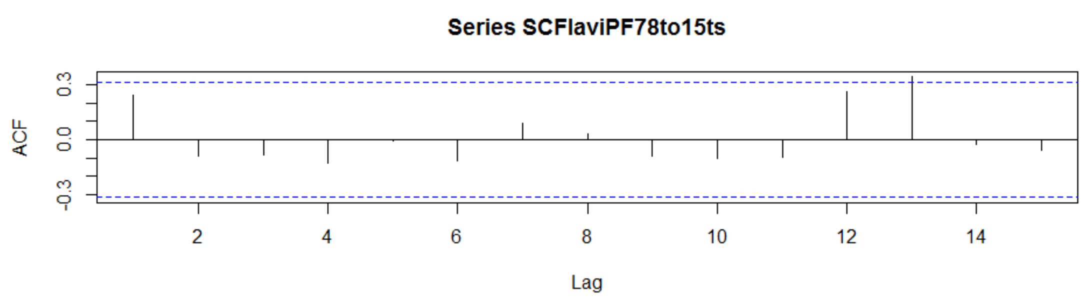

```r
pacf(diff(log(SCFlaviPF78to15ts+1)))
```



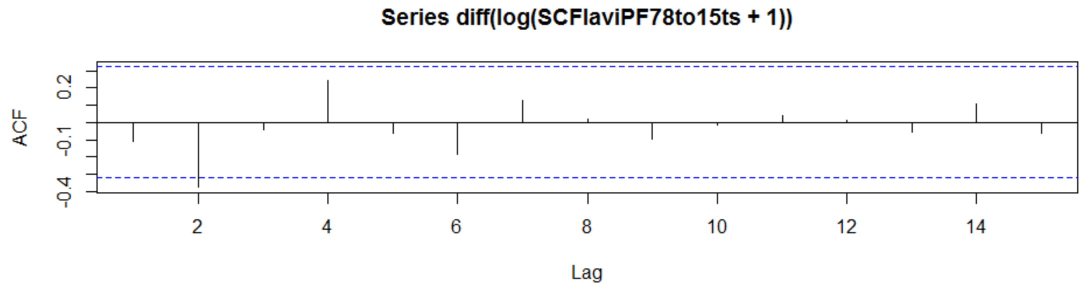
SLE

```r
cat("Calculated differences needed for stationary via KPSS: ",
    ndiffs(log(SCFlaviPF78to15ts+1), alpha=0.05, test="kpss"))

Calculated differences needed for stationary via KPSS:  0
```

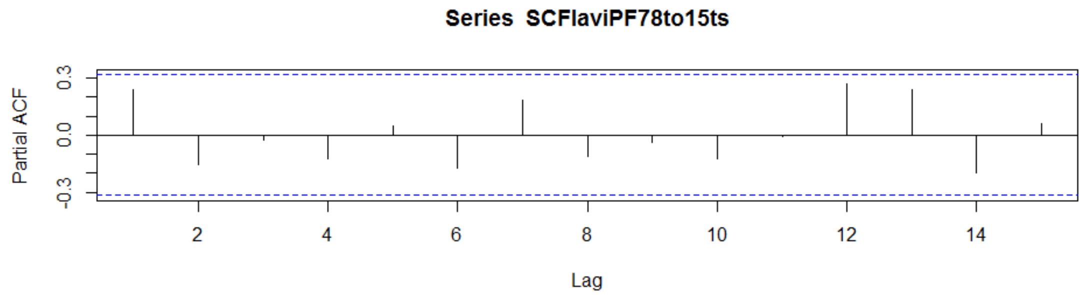
```
cat("\n")

Box.test(log(SCFlaviPF78to15ts+1), type = "Ljung-Box")


    Box-Ljung test

data:  log(SCFlaviPF78to15ts + 1)
X-squared = 2.29, df = 1, p-value = 0.1302

adf.test(log(SCFlaviPF78to15ts+1), alternative = "stationary")


    Augmented Dickey-Fuller Test

data:  log(SCFlaviPF78to15ts + 1)
Dickey-Fuller = -2.9759, Lag order = 3, p-value = 0.1919
alternative hypothesis: stationary

adf.test(diff(log(SCFlaviPF78to15ts+1)), alternative = "stationary")


    Augmented Dickey-Fuller Test

data:  diff(log(SCFlaviPF78to15ts + 1))
Dickey-Fuller = -4.0073, Lag order = 3, p-value = 0.02032
alternative hypothesis: stationary

kpss.test(log(SCFlaviPF78to15ts+1))


    KPSS Test for Level Stationarity

data:  log(SCFlaviPF78to15ts + 1)
KPSS Level = 0.3425, Truncation lag parameter = 1, p-value = 0.1

kpss.test(diff(log(SCFlaviPF78to15ts+1)))


    KPSS Test for Level Stationarity

data:  diff(log(SCFlaviPF78to15ts + 1))
KPSS Level = 0.06706, Truncation lag parameter = 1, p-value = 0.1

acf(SCFlaviPF78to15ts)
```



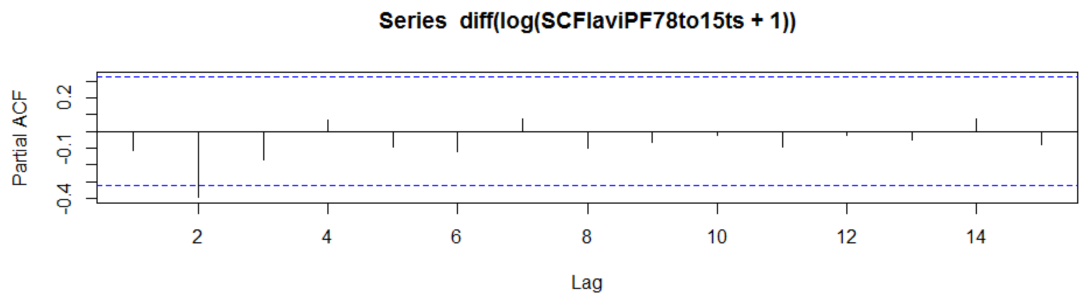

```
acf(diff(log(SCFlaviPF78to15ts+1)))
```




```r
pacf(SCFlaviPF78to15ts)
```




```r
pacf(diff(log(SCFlaviPF78to15ts+1)))
```



WN

```r
cat("Calculated differences needed for stationary via KPSS: ",
    ndiffs(log(SCWNPF01to15ts+1), alpha=0.05, test="kpss"))
```

```
Calculated differences needed for stationary via KPSS:  0
```

```r
cat("\n")
```

```r
Box.test(log(SCWNPF01to15ts+1), type = "Ljung-Box")
```

```

	Box-Ljung test

data:  log(SCWNPF01to15ts + 1)
X-squared = 4.8561, df = 1, p-value = 0.02755
```

```r
adf.test(log(SCWNPF01to15ts+1), alternative = "stationary")
```

```

	Augmented Dickey-Fuller Test

data:  log(SCWNPF01to15ts + 1)
```

```
Dickey-Fuller = -2.4577, Lag order = 2, p-value = 0.398
alternative hypothesis: stationary
```

```r
adf.test(diff(log(SCWNPF01to15ts+1)), alternative = "stationary")
```

```

    Augmented Dickey-Fuller Test

data:  diff(log(SCWNPF01to15ts + 1))
Dickey-Fuller = -1.8499, Lag order = 2, p-value = 0.6296
alternative hypothesis: stationary
```

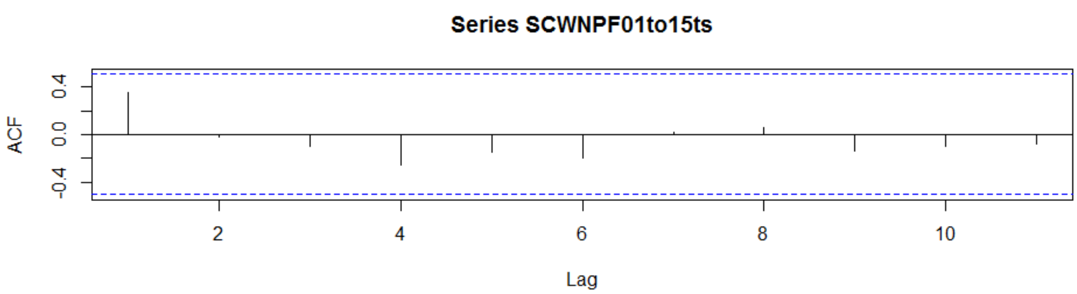
```r
kpss.test(log(SCWNPF01to15ts+1))
```

```

    KPSS Test for Level Stationarity

data:  log(SCWNPF01to15ts + 1)
KPSS Level = 0.20022, Truncation lag parameter = 0, p-value = 0.1
```

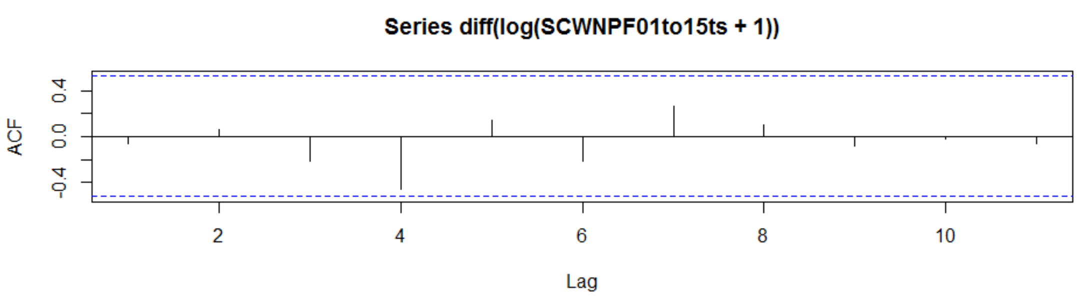
```r
kpss.test(diff(log(SCWNPF01to15ts+1)))
```

```

    KPSS Test for Level Stationarity

data:  diff(log(SCWNPF01to15ts + 1))
KPSS Level = 0.088582, Truncation lag parameter = 0, p-value = 0.1
```

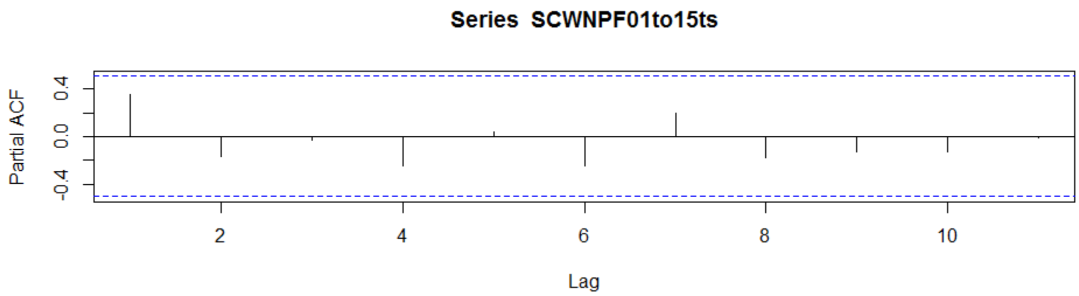
```r
acf(SCWNPF01to15ts)
```



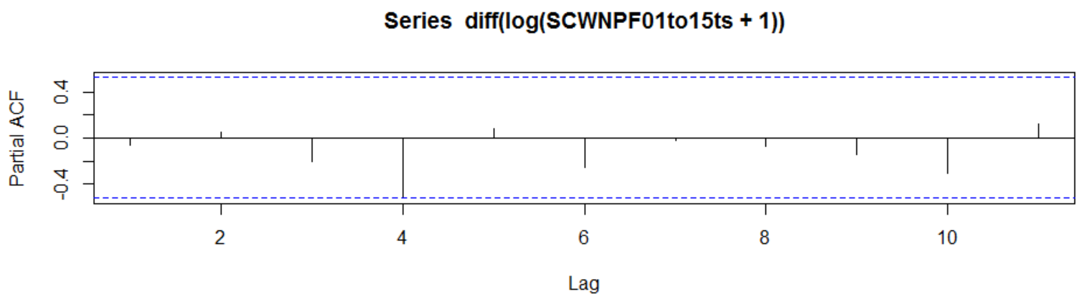

```r
acf(diff(log(SCWNPF01to15ts+1)))
```




```r
pacf(SCWNPF01to15ts)
```




```r
pacf(diff(log(SCWNPF01to15ts+1)))
```



Model Fitting

ARIMA models are fit using auto.arima function in the forecast package. Fitted models are then used to prewhiten data in the Analysis phase. Models only need to be fiited to the BBS data.

BBS Models

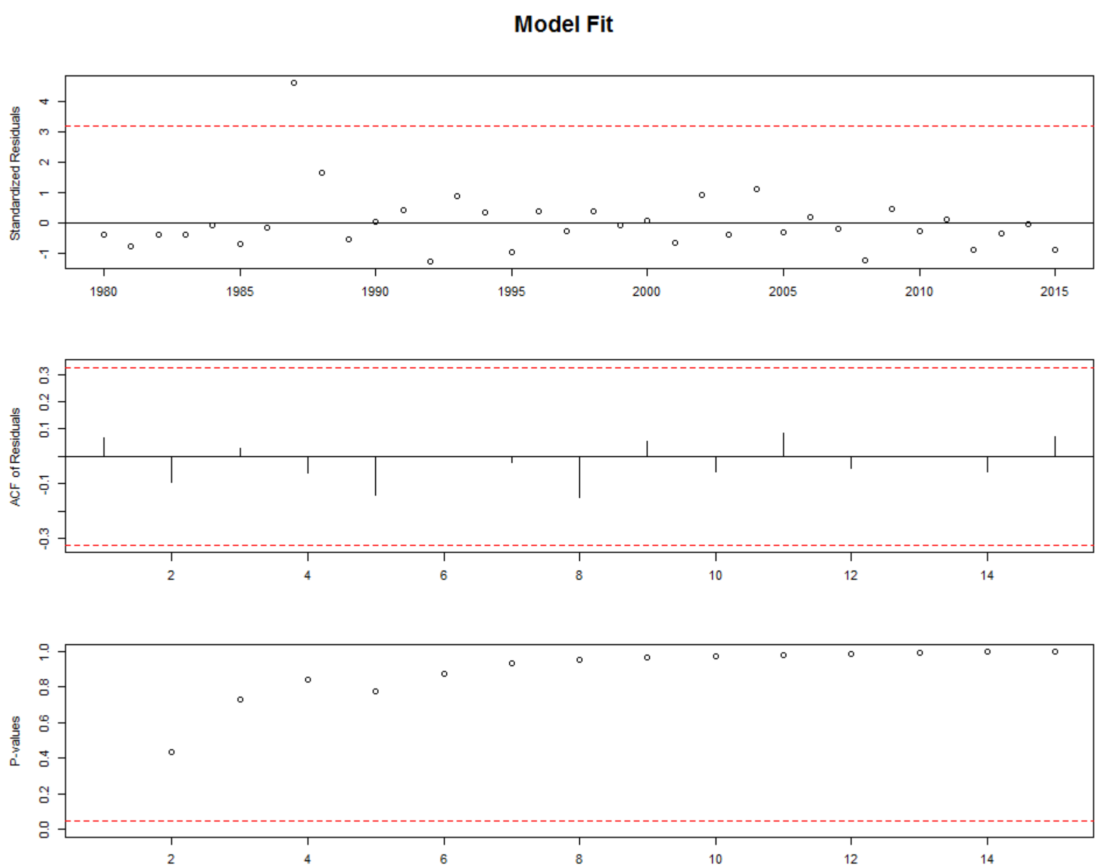
Blue Jay Averages

```r
fitbj <- auto.arima(BJAveragePF78to15ts, seasonal = FALSE, trace = TRUE)


ARIMA(2,1,2) with drift         : Inf
ARIMA(0,1,0) with drift         : 214.8543
ARIMA(1,1,0) with drift         : 212.9345
ARIMA(0,1,1) with drift         : 209.5992
ARIMA(0,1,0)                    : 212.6157
ARIMA(1,1,1) with drift         : Inf
ARIMA(0,1,2) with drift         : 211.2073
ARIMA(1,1,2) with drift         : Inf
ARIMA(0,1,1)                    : 207.2474
ARIMA(1,1,1)                    : 208.8307
ARIMA(0,1,2)                    : 208.7959
ARIMA(1,1,2)                    : Inf

Best model: ARIMA(0,1,1)

fitbj
```

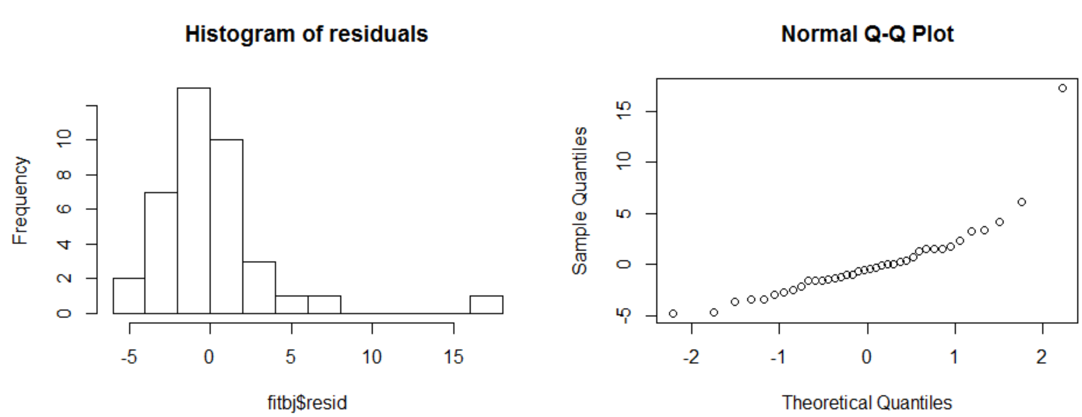
```
Series: BJAveragePF78to15ts
ARIMA(0,1,1)

Coefficients:
          ma1
      -0.5587
s.e.   0.1725

```

```
sigma^2 estimated as 13.95:  log likelihood=-101.45
AIC=206.89   AICc=207.25   BIC=210.12
```

```
par(oma=c(0,0,2,0))
tsdiag(Arima(BJAveragePF78to15ts, model=fitbj))
title("Model Fit", outer = TRUE)
```




```
par(mfrow=c(1,2))
hist(fitbj$resid, nclass="FD", main="Histogram of residuals")
qqnorm(fitbj$resid)
```

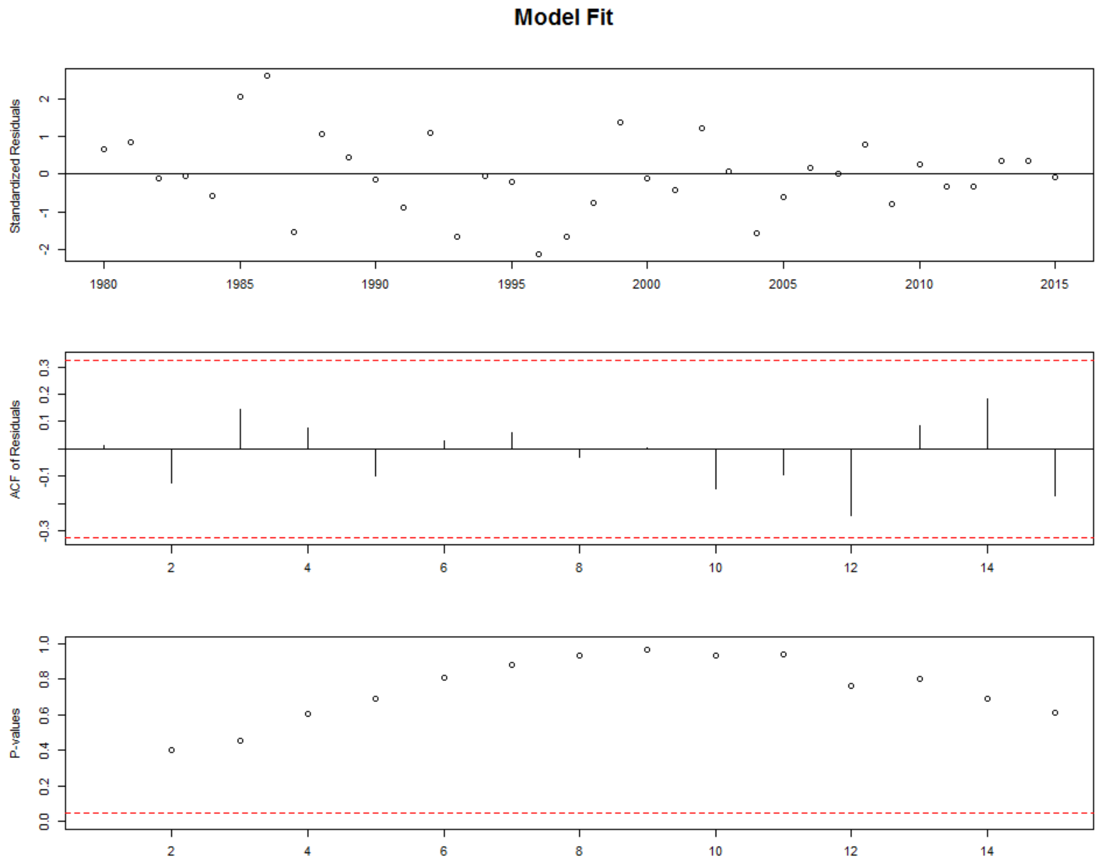



```
Box.test(fitbj$resid, type = "Ljung-Box")

	Box-Ljung test

```

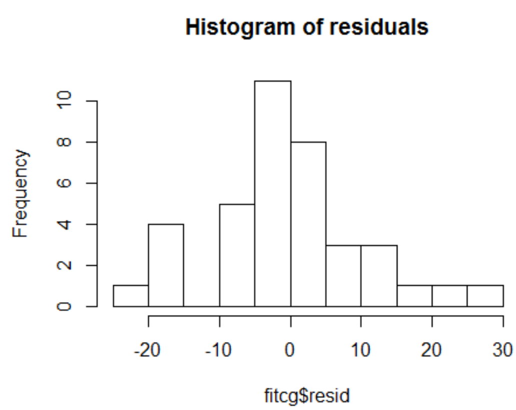
```
data:  fitbj$resid
X-squared = 0.14372, df = 1, p-value = 0.7046
```

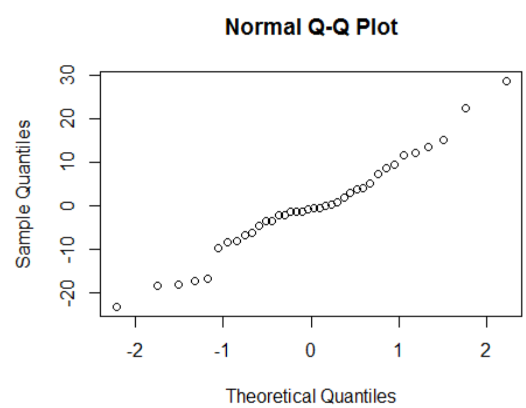
Common Grackle Averages

```
fitcg=auto.arima(CGAveragePF78to15ts, seasonal=FALSE, trace=TRUE)


ARIMA(2,1,2) with drift         : 295.9334
ARIMA(0,1,0) with drift         : 295.6592
ARIMA(1,1,0) with drift         : 293.196
ARIMA(0,1,1) with drift         : 289.3978
ARIMA(0,1,0)                    : 293.425
ARIMA(1,1,1) with drift         : 291.8666
ARIMA(0,1,2) with drift         : 291.808
ARIMA(1,1,2) with drift         : 294.0616
ARIMA(0,1,1)                    : 287.0651
ARIMA(1,1,1)                    : 289.395
ARIMA(0,1,2)                    : 289.3463
ARIMA(1,1,2)                    : 291.4329

Best model: ARIMA(0,1,1)

fitcg

Series: CGAveragePF78to15ts
ARIMA(0,1,1)

Coefficients:
          ma1
      -0.5297
s.e.   0.1371

sigma^2 estimated as 120.8:  log likelihood=-141.36
AIC=286.71   AICc=287.07   BIC=289.93

par(oma=c(0,0,2,0))
tsdiag(Arima(CGAveragePF78to15ts, model=fitcg))
title("Model Fit", outer = TRUE)
```




```r
par(mfrow=c(1,2))
hist(fitcg$resid, nclass="FD", main="Histogram of residuals")
qqnorm(fitcg$resid)
```





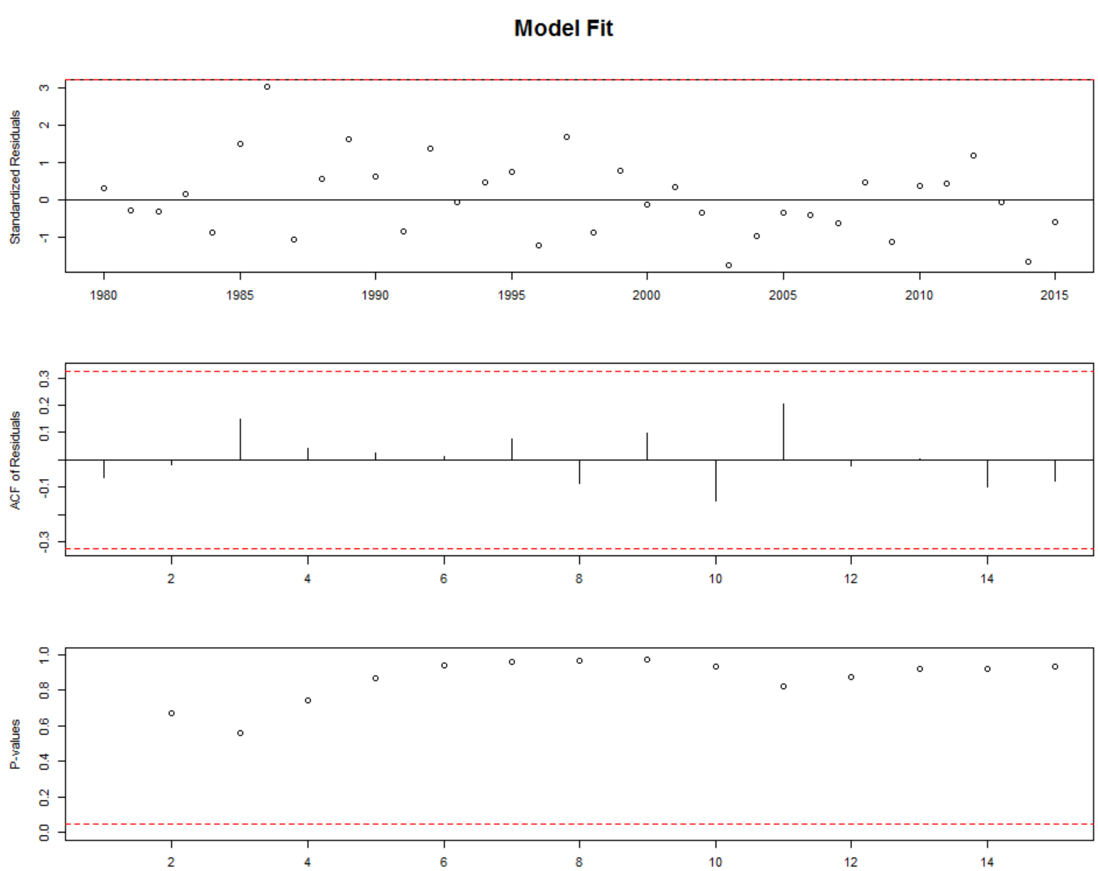

```r
Box.test(fitcg$resid, type = "Ljung-Box")

	Box-Ljung test

data:  fitcg$resid
X-squared = 0.0023868, df = 1, p-value = 0.961
```

Mourning Dove Averages

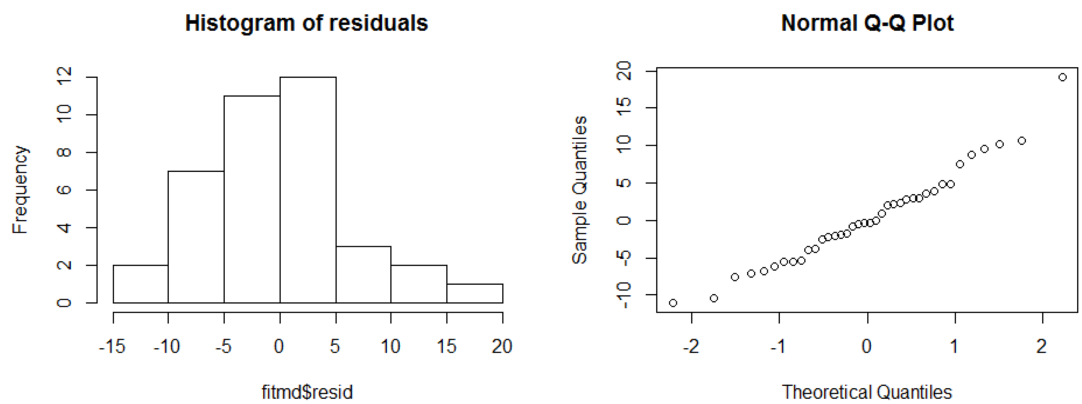
```r
fitmd=auto.arima(MDAveragePF78to15ts, seasonal=FALSE, trace=TRUE)
```

```
 ARIMA(2,1,2) with drift         : 255.5328
 ARIMA(0,1,0) with drift         : 255.9683
 ARIMA(1,1,0) with drift         : 250.7897
 ARIMA(0,1,1) with drift         : 248.8007
 ARIMA(0,1,0)                    : 253.7423
 ARIMA(1,1,1) with drift         : 251.1112
 ARIMA(0,1,2) with drift         : 250.9443
 ARIMA(1,1,2) with drift         : 253.1824
 ARIMA(0,1,1)                    : 246.5557
 ARIMA(1,1,1)                    : 248.6943
 ARIMA(0,1,2)                    : 248.5024
 ARIMA(1,1,2)                    : 250.5361

 Best model: ARIMA(0,1,1)

 fitmd

Series: MDAveragePF78to15ts
ARIMA(0,1,1)

Coefficients:
          ma1
      -0.4929
s.e.   0.1254

sigma^2 estimated as 40.47:  log likelihood=-121.1
AIC=246.2   AICc=246.56   BIC=249.42
```

```
 par(oma=c(0,0,2,0))
 tsdiag(Arima(MDAveragePF78to15ts, model=fitmd))
 title("Model Fit", outer = TRUE)
```



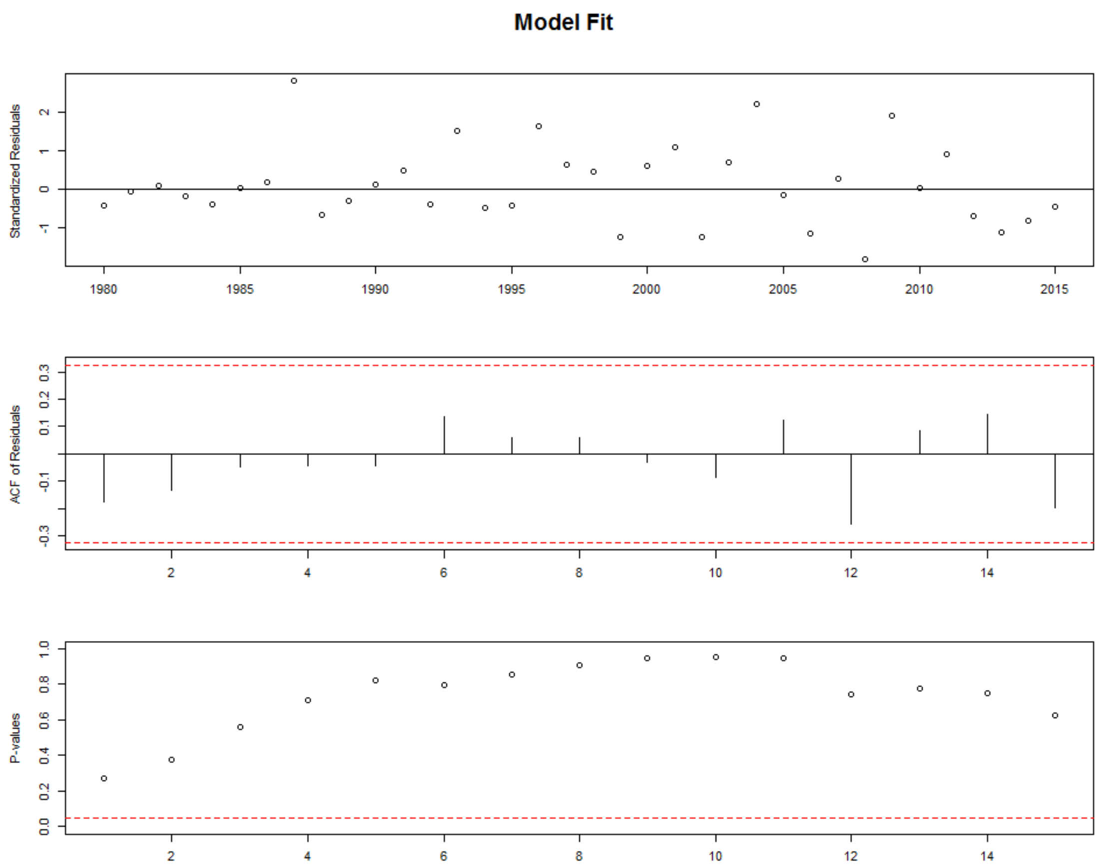
```r
par(mfrow=c(1,2))
hist(fitmd$resid, nclass="FD", main="Histogram of residuals")
qqnorm(fitmd$resid)
```



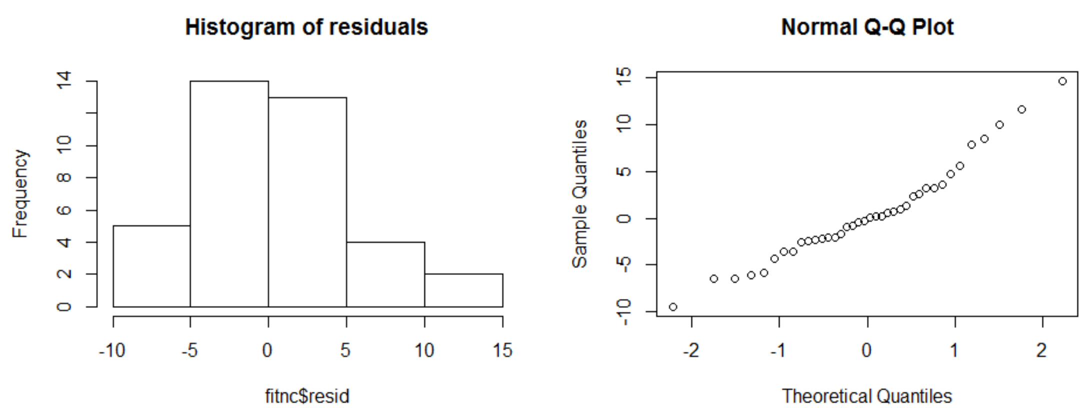

```r
Box.test(fitmd$resid, type = "Ljung-Box")


	Box-Ljung test

data:  fitmd$resid
X-squared = 0.17061, df = 1, p-value = 0.6796
```

Northern Cardinal Averages

```r
fitnc=auto.arima(NCAveragePF78to15ts, seasonal=FALSE, trace=TRUE)


 ARIMA(2,1,2) with drift         : Inf
 ARIMA(0,1,0) with drift         : 231.5002
 ARIMA(1,1,0) with drift         : 232.7511
 ARIMA(0,1,1) with drift         : 232.1039
 ARIMA(0,1,0)                    : 229.5898
 ARIMA(1,1,1) with drift         : Inf

 Best model: ARIMA(0,1,0)

fitnc

Series: NCAveragePF78to15ts
ARIMA(0,1,0)

sigma^2 estimated as 27.39:  log likelihood=-113.74
AIC=229.48   AICc=229.59   BIC=231.09

par(oma=c(0,0,2,0))
tsdiag(Arima(NCAveragePF78to15ts, model=fitnc))
title("Model Fit", outer = TRUE)
```




```r
par(mfrow=c(1,2))
hist(fitnc$resid, nclass="FD", main="Histogram of residuals")
qqnorm(fitnc$resid)
```




```r
Box.test(fitnc$resid, type = "Ljung-Box")

	Box-Ljung test

data:  fitnc$resid
X-squared = 1.243, df = 1, p-value = 0.2649
```

Northern Mockingbird Averages

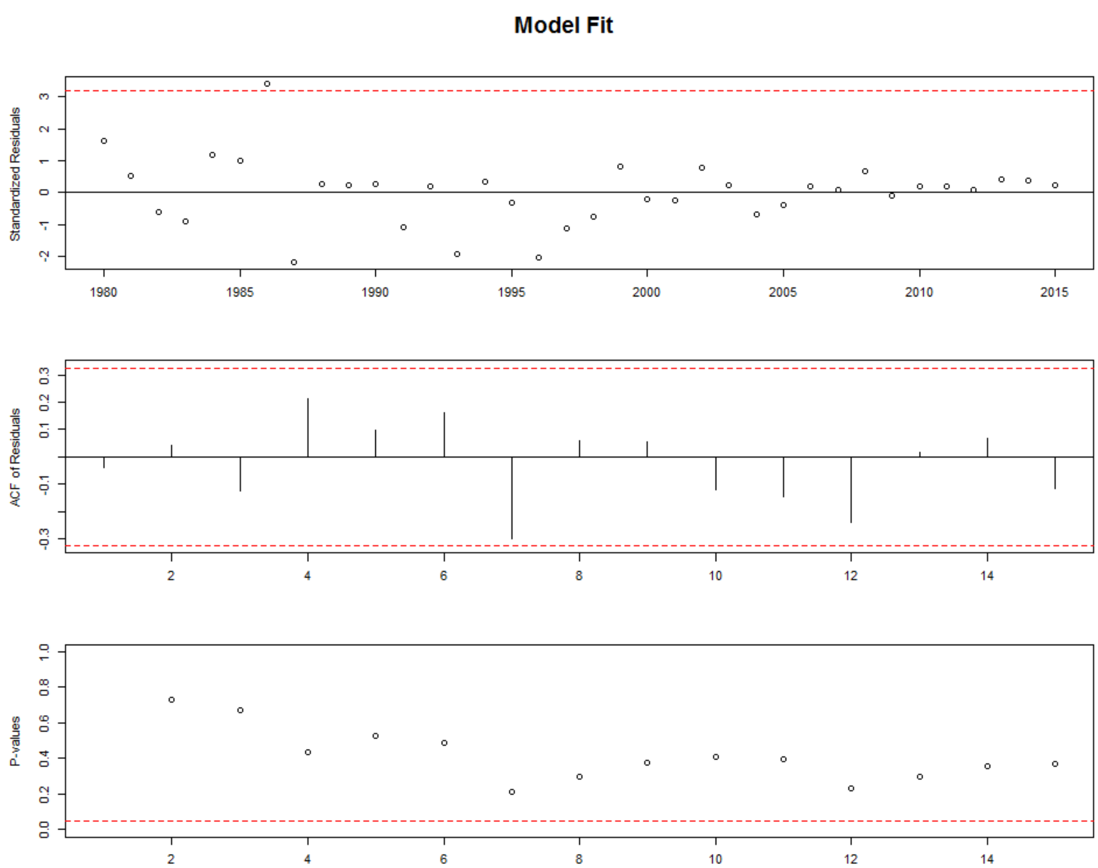
```r
fitnm=auto.arima(NMAveragePF78to15ts, seasonal=FALSE, trace=TRUE)
```

```
ARIMA(2,1,2) with drift         : 343.1423
ARIMA(0,1,0) with drift         : 345.482
ARIMA(1,1,0) with drift         : 338.8013
ARIMA(0,1,1) with drift         : 336.1272
ARIMA(0,1,0)                    : 343.8001
ARIMA(1,1,1) with drift         : 338.5844
ARIMA(0,1,2) with drift         : 338.5816
ARIMA(1,1,2) with drift         : Inf
ARIMA(0,1,1)                    : 337.3919

Best model: ARIMA(0,1,1) with drift
```

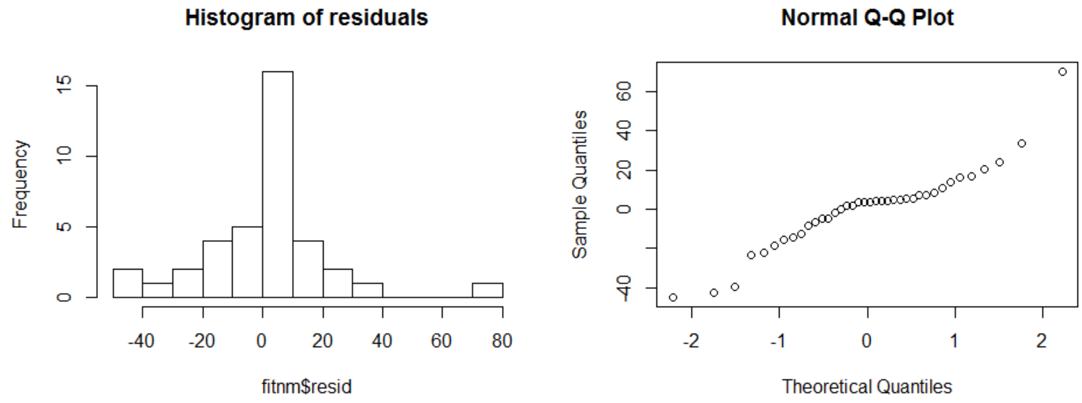
```
fitnm
```

```
Series: NMAveragePF78to15ts
ARIMA(0,1,1) with drift

Coefficients:
          ma1    drift
      -0.5898  -3.3783
s.e.   0.1289   1.4587

sigma^2 estimated as 425.5:  log likelihood=-164.7
AIC=335.4   AICc=336.13   BIC=340.23
```

```
par(oma=c(0,0,2,0))
tsdiag(Arima(NMAveragePF78to15ts, model=fitnm))
title("Model Fit", outer = TRUE)
```



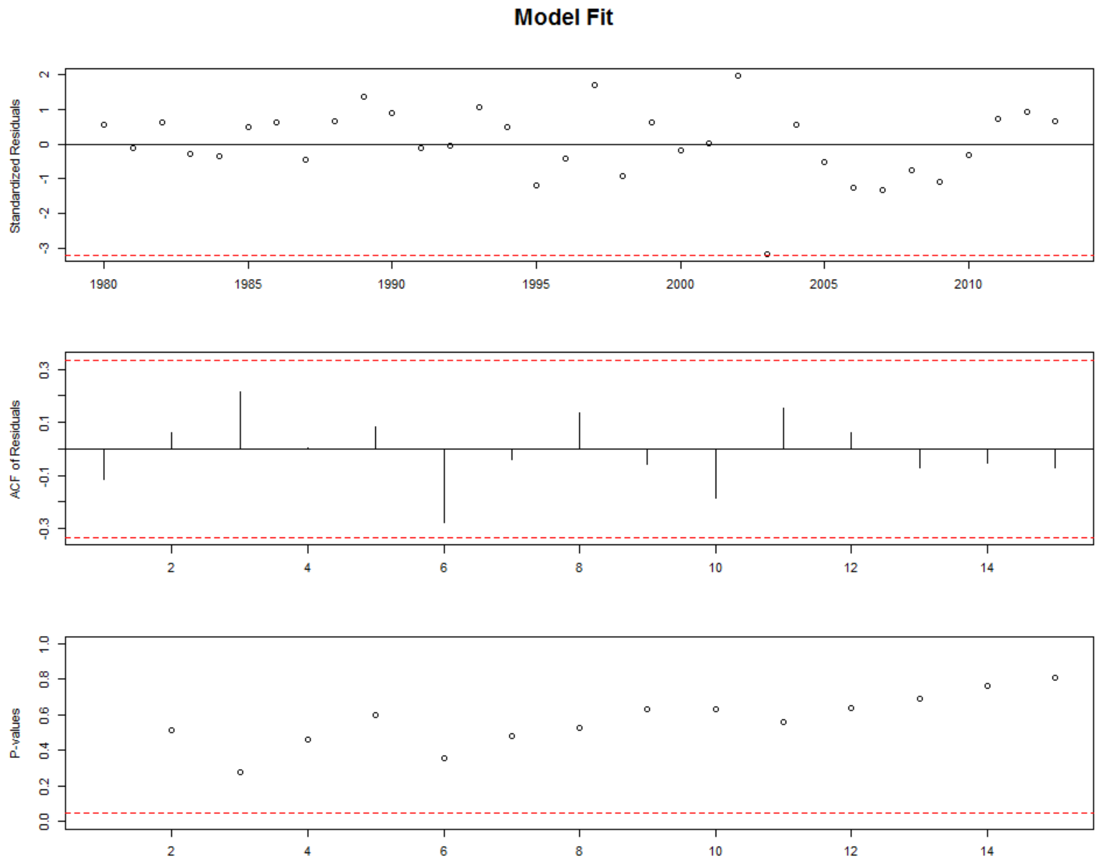

```
par(mfrow=c(1,2))
hist(fitnm$resid, nclass="FD", main="Histogram of residuals")
qqnorm(fitnm$resid)
```



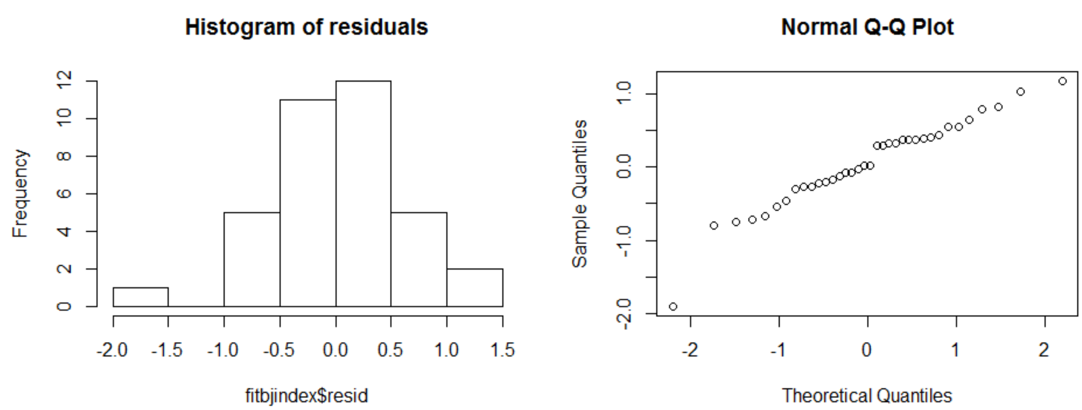

```
Box.test(fitnm$resid, type = "Ljung-Box")

    Box-Ljung test

data:  fitnm$resid
X-squared = 0.038131, df = 1, p-value = 0.8452
```

Blue Jay Trend Index

```
fitbjindex <- auto.arima(BJIndexPF78to13ts, seasonal = FALSE, trace = TRUE)


 ARIMA(2,1,2) with drift         : Inf
 ARIMA(0,1,0) with drift         : 90.43102
 ARIMA(1,1,0) with drift         : 79.79555
 ARIMA(0,1,1) with drift         : 72.04334
 ARIMA(0,1,0)                    : 89.46693
 ARIMA(1,1,1) with drift         : 73.65549
 ARIMA(0,1,2) with drift         : Inf
 ARIMA(1,1,2) with drift         : Inf
 ARIMA(0,1,1)                    : 82.34281

 Best model: ARIMA(0,1,1) with drift

fitbjindex

Series: BJIndexPF78to13ts
ARIMA(0,1,1) with drift

Coefficients:
          ma1    drift
      -0.8432  -0.1913
s.e.   0.1499   0.0204

sigma^2 estimated as 0.3648:  log likelihood=-32.63
AIC=71.27   AICc=72.04   BIC=75.94

par(oma=c(0,0,2,0))
tsdiag(Arima(BJIndexPF78to13ts, model=fitbjindex))
title("Model Fit", outer = TRUE)
```




```
par(mfrow=c(1,2))
hist(fitbjindex$resid, nclass="FD", main="Histogram of residuals")
qqnorm(fitbjindex$resid)
```

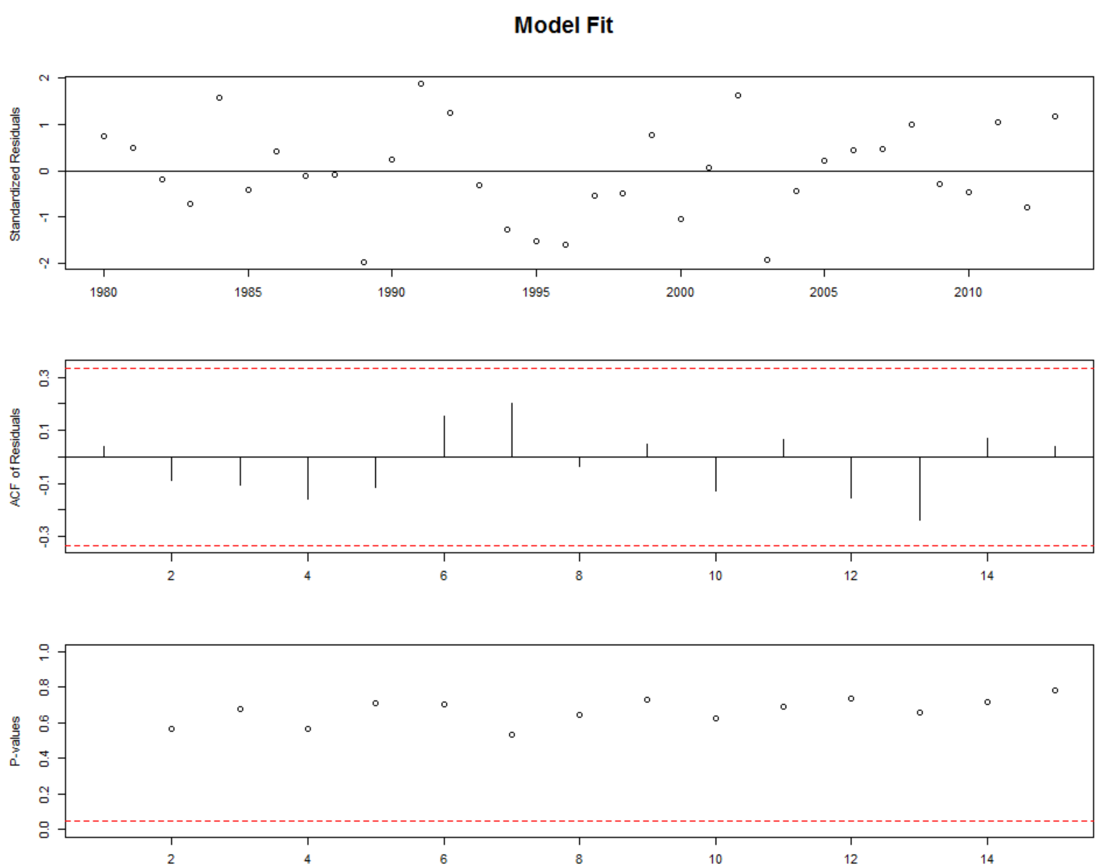



```
Box.test(fitbjindex$resid, type = "Ljung-Box")


    Box-Ljung test

data:  fitbjindex$resid
X-squared = 0.33475, df = 1, p-value = 0.5629
```

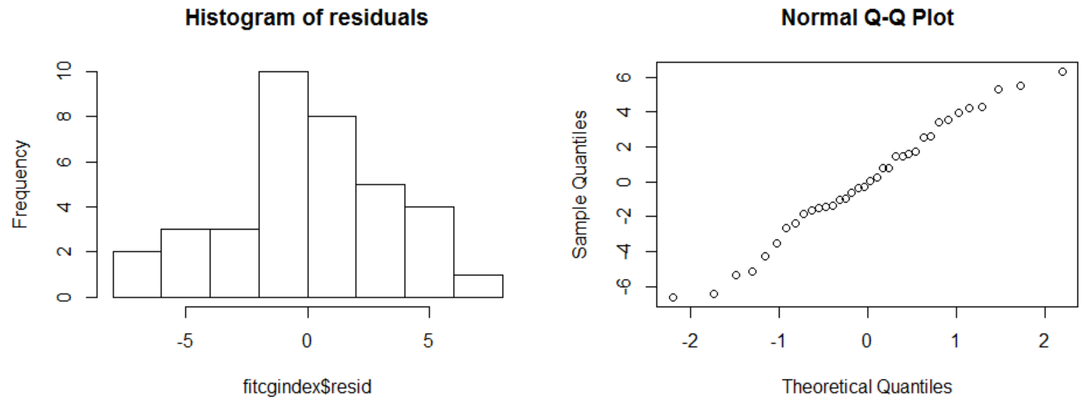
Common Grackle Trend Index

```
fitcgindex <- auto.arima(CGIndexPF78to13ts, seasonal = FALSE, trace = TRUE)
```

```
 ARIMA(2,1,2) with drift         : Inf
 ARIMA(0,1,0) with drift         : 191.9144
 ARIMA(1,1,0) with drift         : 192.0968
 ARIMA(0,1,1) with drift         : 191.2062
 ARIMA(0,1,0)                    : 191.9284
 ARIMA(1,1,1) with drift         : Inf
 ARIMA(0,1,2) with drift         : 192.9716
 ARIMA(1,1,2) with drift         : Inf
 ARIMA(0,1,1)                    : 193.4492

 Best model: ARIMA(0,1,1) with drift

 fitcgindex

Series: CGIndexPF78to13ts
ARIMA(0,1,1) with drift

Coefficients:
          ma1    drift
      -0.3834  -1.0107
s.e.   0.2171   0.3617

sigma^2 estimated as 11.33:  log likelihood=-92.22
AIC=190.43   AICc=191.21   BIC=195.1

 par(oma=c(0,0,2,0))
 tsdiag(Arima(CGIndexPF78to13ts, model=fitcgindex))
 title("Model Fit", outer = TRUE)
```




```
 par(mfrow=c(1,2))
 hist(fitcgindex$resid, nclass="FD", main="Histogram of residuals")
 qqnorm(fitcgindex$resid)
```




```
Box.test(fitcgindex$resid, type = "Ljung-Box")
```

```

    Box-Ljung test

data:  fitcgindex$resid
X-squared = 0.17067, df = 1, p-value = 0.6795
```

Mourning Dove Trend Index

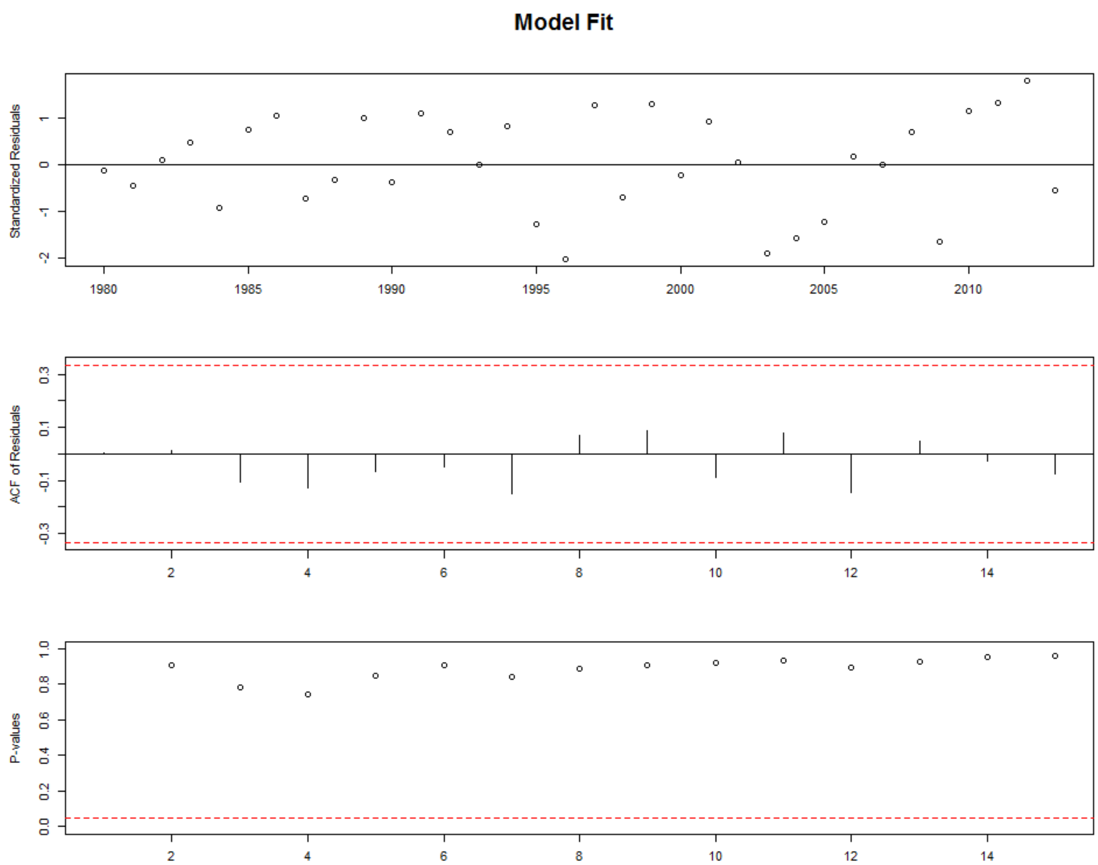
```
fitmdindex=auto.arima(MDIndexPF78to13ts, seasonal=FALSE, trace=TRUE)
```

```

 ARIMA(2,1,2) with drift         : Inf
 ARIMA(0,1,0) with drift         : 204.8869
 ARIMA(1,1,0) with drift         : 203.7522
 ARIMA(0,1,1) with drift         : 204.0641
 ARIMA(0,1,0)                    : 203.9318
 ARIMA(2,1,0) with drift         : 206.2612
 ARIMA(1,1,1) with drift         : 206.2448
 ARIMA(2,1,1) with drift         : 208.9777
 ARIMA(1,1,0)                    : 203.8543

 Best model: ARIMA(1,1,0) with drift
```

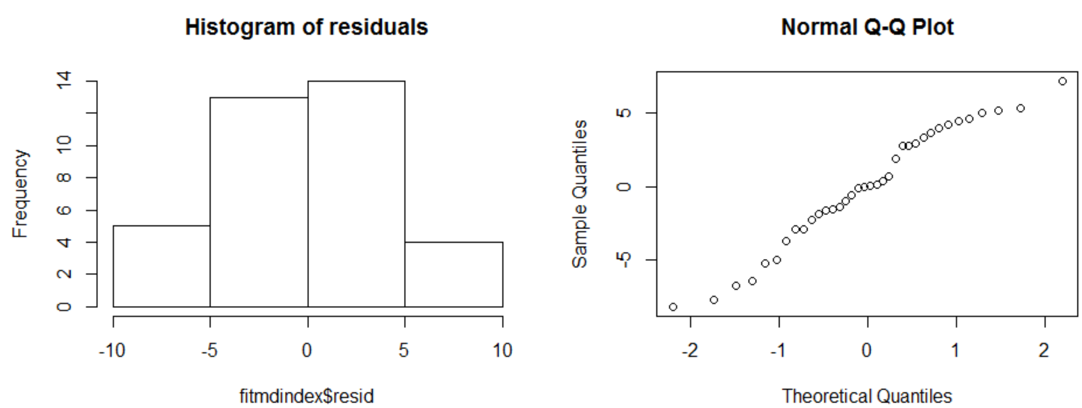
```
fitmdindex
```

```
Series: MDIndexPF78to13ts 
ARIMA(1,1,0) with drift         

Coefficients:
          ar1   drift
      -0.3104  0.8647
s.e.   0.1604  0.5236

sigma^2 estimated as 16.24:  log likelihood=-98.49
AIC=202.98   AICc=203.75   BIC=207.64
```

```
par(oma=c(0,0,2,0))
tsdiag(Arima(MDIndexPF78to13ts, model=fitmdindex))
title("Model Fit", outer = TRUE)
```




```r
par(mfrow=c(1,2))
hist(fitmdindex$resid, nclass="FD", main="Histogram of residuals")
qqnorm(fitmdindex$resid)
```

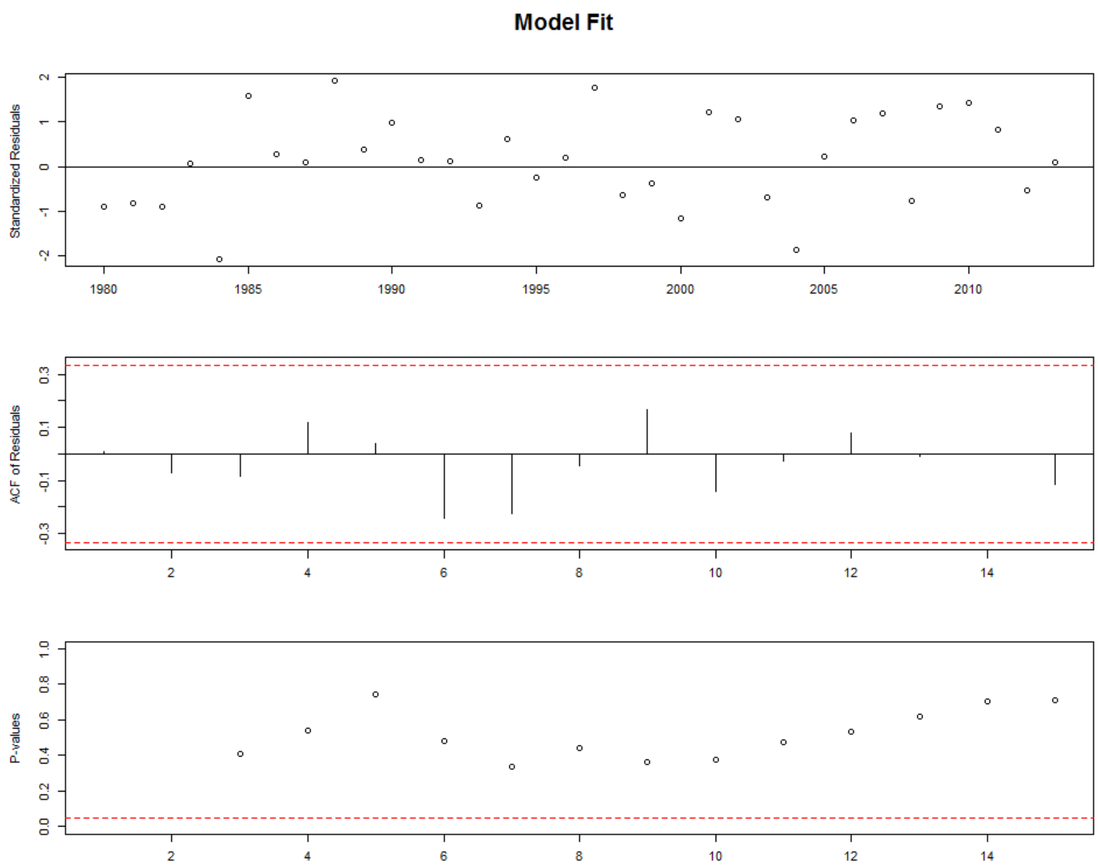



```r
Box.test(fitmdindex$resid, type = "Ljung-Box")


    Box-Ljung test

data:  fitmdindex$resid
X-squared = 0.00069059, df = 1, p-value = 0.979
```

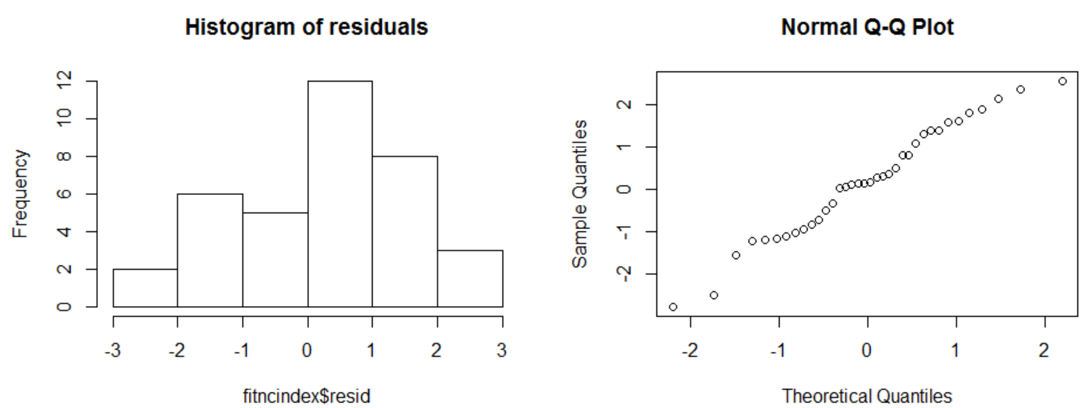
Northern Cardinal Trend Index

```r
fitncindex=auto.arima(NCIndexPF78to13ts, seasonal=FALSE, trace=TRUE)
```

```
 ARIMA(2,1,2) with drift         : Inf
 ARIMA(0,1,0) with drift         : 144.83
 ARIMA(1,1,0) with drift         : 130.5015
 ARIMA(0,1,1) with drift         : Inf
 ARIMA(0,1,0)                    : 142.7413
 ARIMA(2,1,0) with drift         : 130.4979
 ARIMA(2,1,1) with drift         : Inf
 ARIMA(3,1,1) with drift         : 134.4352
 ARIMA(2,1,0)                    : 128.7578
 ARIMA(1,1,0)                    : 128.5575
 ARIMA(1,1,1)                    : 128.1251
 ARIMA(2,1,2)                    : 132.0186
 ARIMA(1,1,1) with drift         : 129.1125
 ARIMA(0,1,1)                    : 128.048
 ARIMA(0,1,2)                    : 127.4672
 ARIMA(1,1,3)                    : 132.2656
 ARIMA(0,1,2) with drift         : 129.1828
 ARIMA(1,1,2)                    : 129.6167
 ARIMA(0,1,3)                    : 129.5481

 Best model: ARIMA(0,1,2)

 fitncindex

Series: NCIndexPF78to13ts
ARIMA(0,1,2)

Coefficients:
          ma1     ma2
      -0.8566  0.3766
s.e.   0.1848  0.2356

sigma^2 estimated as 1.8:  log likelihood=-60.35
AIC=126.69   AICc=127.47   BIC=131.36

 par(oma=c(0,0,2,0))
 tsdiag(Arima(NCIndexPF78to13ts, model=fitncindex))
 title("Model Fit", outer = TRUE)
```




```
par(mfrow=c(1,2))
hist(fitncindex$resid, nclass="FD", main="Histogram of residuals")
qqnorm(fitncindex$resid)
```




```
Box.test(fitncindex$resid, type = "Ljung-Box")

    Box-Ljung test

data:  fitncindex$resid
X-squared = 0.0017779, df = 1, p-value = 0.9664
```

Northern Mockingbird Trend Index

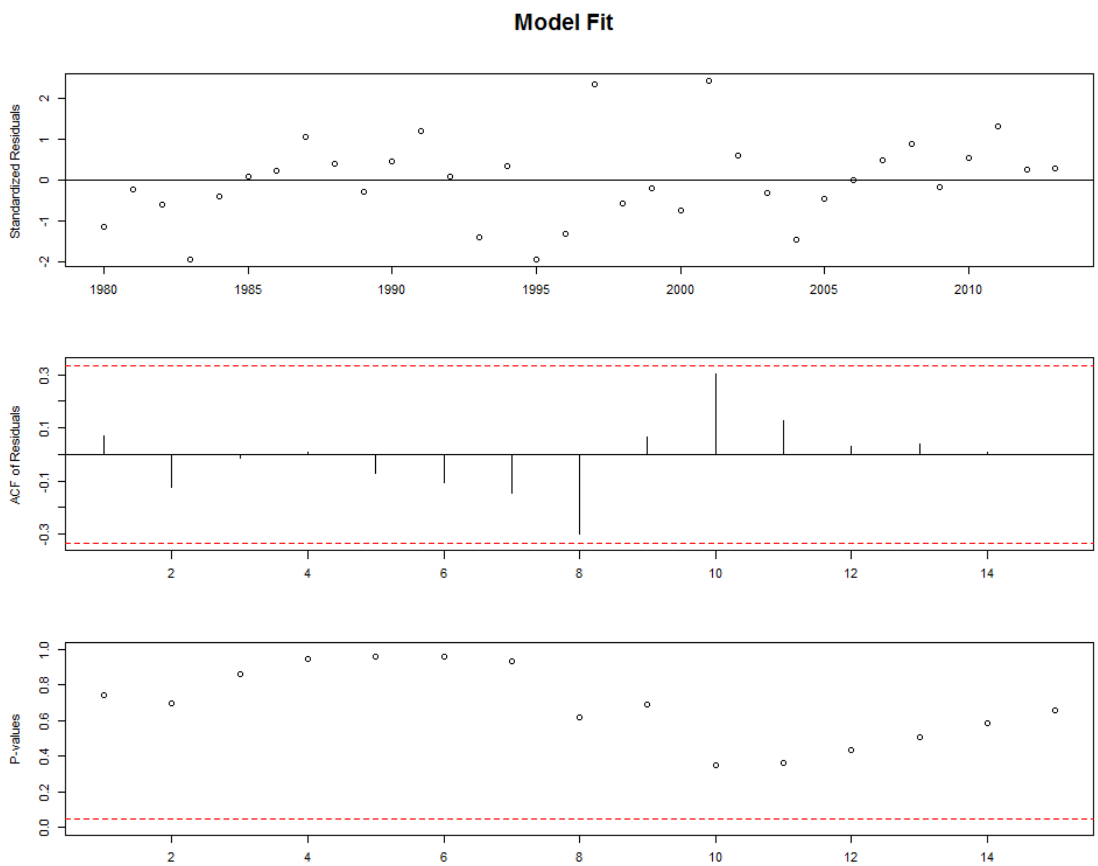
```
fitnmindex=auto.arima(NMIndexPF78to13ts, seasonal=FALSE, trace=TRUE)
```

```

ARIMA(2,1,2) with drift         : Inf
ARIMA(0,1,0) with drift         : 133.8833
ARIMA(1,1,0) with drift         : 136.187
ARIMA(0,1,1) with drift         : 136.1553
ARIMA(0,1,0)                    : 153.1394
ARIMA(1,1,1) with drift         : 138.4137

Best model: ARIMA(0,1,0) with drift
```

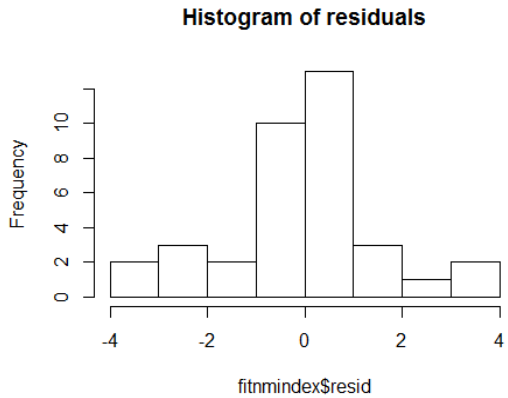
```
fitnmindex
```

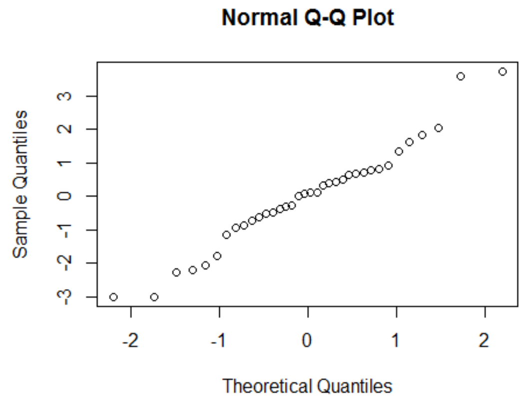
```
Series: NMIndexPF78to13ts
ARIMA(0,1,0) with drift

Coefficients:
        drift
      -1.4180
s.e.   0.2602

sigma^2 estimated as 2.369:  log likelihood=-64.75
AIC=133.51   AICc=133.88   BIC=136.62
```

```
par(oma=c(0,0,2,0))
tsdiag(Arima(NMIndexPF78to13ts, model=fitnmindex))
title("Model Fit", outer = TRUE)
```




```
par(mfrow=c(1,2))
hist(fitnmindex$resid, nclass="FD", main="Histogram of residuals")
qqnorm(fitnmindex$resid)
```






```
Box.test(fitnmindex$resid, type = "Ljung-Box")


    Box-Ljung test

data:  fitnmindex$resid
X-squared = 0.11188, df = 1, p-value = 0.738
```

Transmission Models

Transmission data does not need to be fitted with a models. Models only need to be fitted to the BBS data. BBS data models are then fitted to the Transmission data to prewhiten.

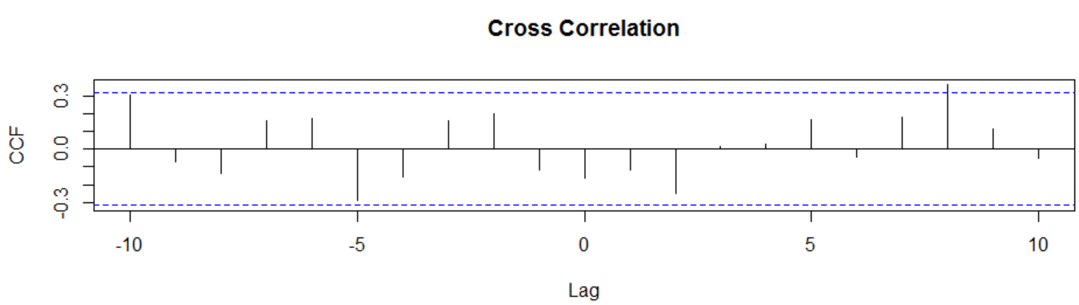
Analysis

ARIMA models fitted to the BBS data are used to prewhiten both the BBS and Transmission data prior to a Cross Correlation Function being calculated for the residuals at each time lag. Significant lags are then checked for Granger Causality.

Cross Correlations

Blue Jay Averages x SLE Sentinel Chicken Seroconversions

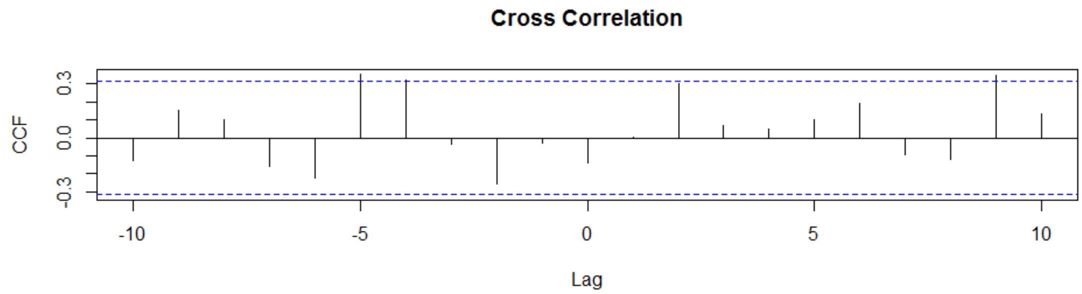
```
resbj<-resid(Arima(BJAveragePF78to15ts, model=fitbj))
resscbj<-resid(Arima(log(SCSLEPF78to15ts+1), model=fitbj))
cat("\n")

ccf(resbj, resscbj, lag.max = 10, type = "correlation", plot = TRUE,
    ylab="CCF", main = "Cross Correlation")
```



Common Grackle Averages x SLE Sentinel Chicken Seroconversions

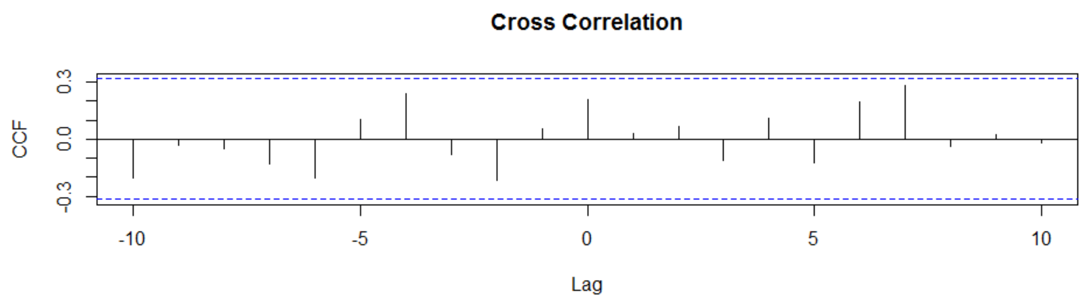
```
rescg<-resid(Arima(CGAveragePF78to15ts, model=fitcg))
ressccg<-resid(Arima(log(SCSLEPF78to15ts+1), model=fitcg))
cat("\n")
```

```r
ccf(rescg, ressccg, lag.max = 10, type = "correlation", plot = TRUE,
    ylab="CCF", main = "Cross Correlation")
```



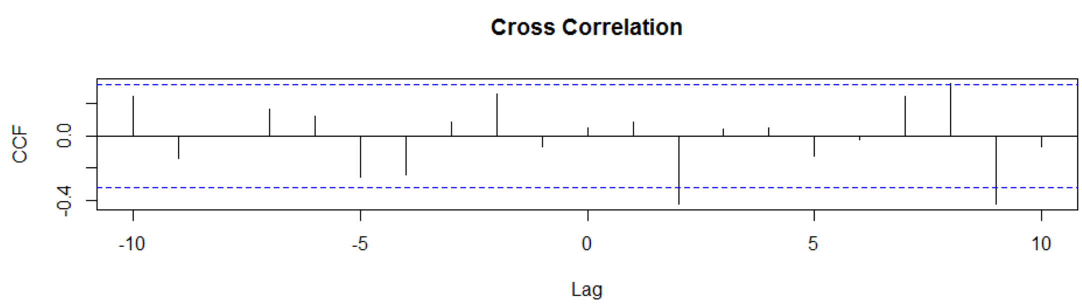
Mourning Dove Averages x SLE Sentinel Chicken Seroconversions

```r
resmd<-resid(Arima(MDAveragePF78to15ts, model=fitmd))
resscmd<-resid(Arima(log(SCSLEPF78to15ts+1), model=fitmd))
cat("\n")

ccf(resmd, resscmd, lag.max = 10, type = "correlation", plot = TRUE,
    ylab="CCF", main = "Cross Correlation")
```



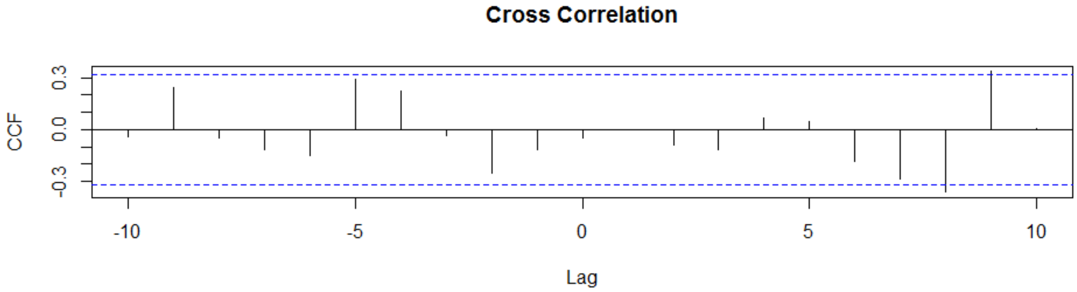
Northern Cardinal Averages x SLE Sentinel Chicken Seroconversions

```r
resnc<-resid(Arima(NCAveragePF78to15ts, model=fitnc))
resscnc<-resid(Arima(log(SCSLEPF78to15ts+1), model=fitnc))
cat("\n")

ccf(resnc, resscnc, lag.max = 10, type = "correlation", plot = TRUE,
    ylab="CCF", main = "Cross Correlation")
```



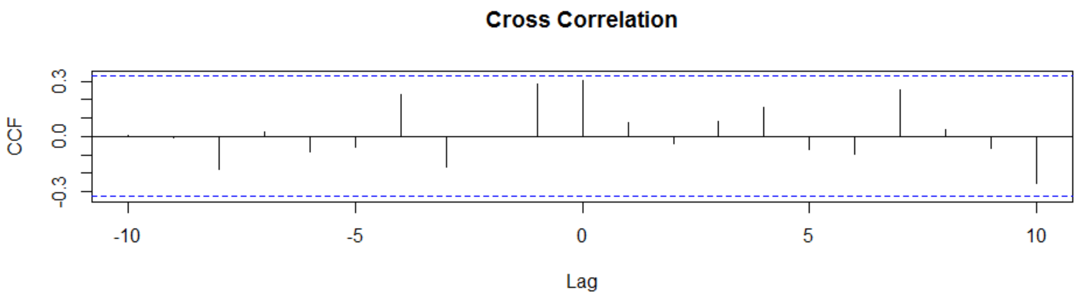
Northern Mockingbird Averages x SLE Sentinel Chicken Seroconversions

```r
resnm<-resid(Arima(NMAveragePF78to15ts, model=fitnm))
resscnm<-resid(Arima(log(SCSLEPF78to15ts+1), model=fitnm))
cat("\n")

ccf(resnm, resscnm, lag.max = 10, type = "correlation", plot = TRUE,
    ylab="CCF", main = "Cross Correlation")
```



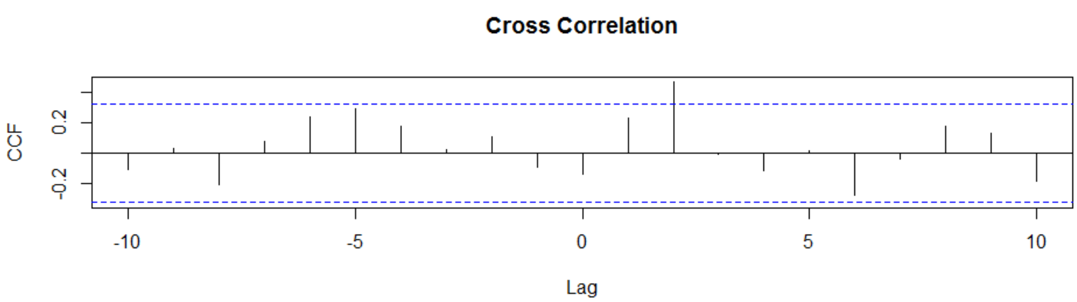
Blue Jay Index x SLE Sentinel Chicken Seroconversions

```
resbji<-resid(Arima(BJIndexPF78to13ts, model=fitbj))
resscbji<-resid(Arima(log(SCSLEPF78to13ts+1), model=fitbj))
cat("\n")

ccf(resbji, resscbji, lag.max = 10, type = "correlation", plot = TRUE,
    ylab="CCF", main = "Cross Correlation")
```



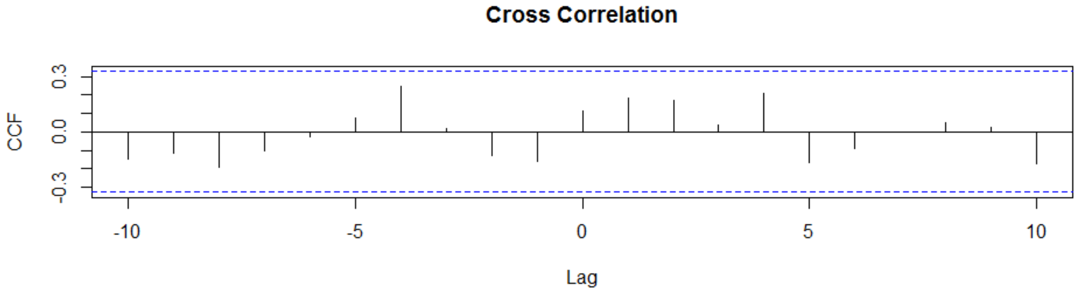
Common Grackle Index x SLE Sentinel Chicken Seroconversions

```
rescgi<-resid(Arima(CGIndexPF78to13ts, model=fitcg))
ressccgi<-resid(Arima(log(SCSLEPF78to13ts+1), model=fitcg))
cat("\n")

ccf(rescgi, ressccgi, lag.max = 10, type = "correlation", plot = TRUE,
    ylab="CCF", main = "Cross Correlation")
```



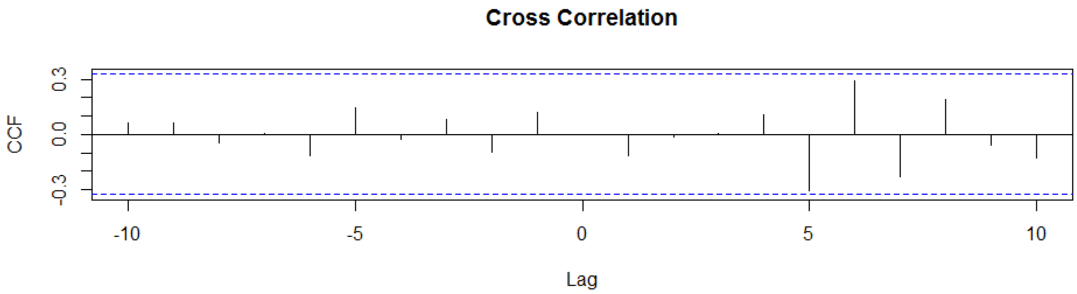
Mourning Dove Index x SLE Sentinel Chicken Seroconversions

```
resmdi<-resid(Arima(MDIndexPF78to13ts, model=fitmd))
resscmdi<-resid(Arima(log(SCSLEPF78to13ts+1), model=fitmd))
cat("\n")

ccf(resmdi, resscmdi, lag.max = 10, type = "correlation", plot = TRUE,
    ylab="CCF", main = "Cross Correlation")
```



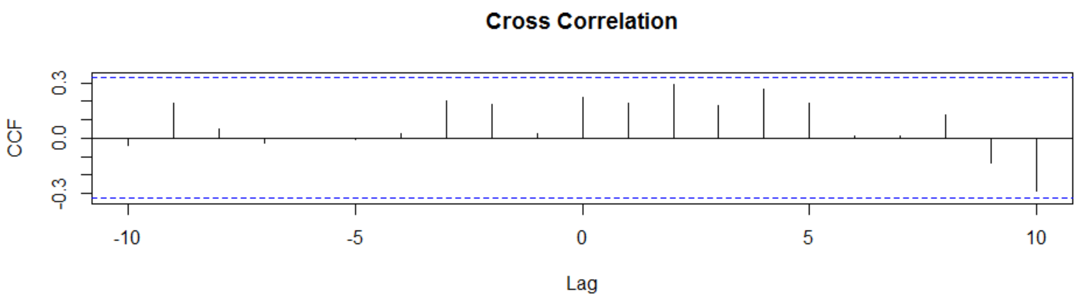
Northern Cardinal Index x SLE Sentinel Chicken Seroconversions

```
resnci<-resid(Arima(NCIndexPF78to13ts, model=fitnc))
resscnci<-resid(Arima(log(SCSLEPF78to13ts+1), model=fitnc))
cat("\n")

ccf(resnci, resscnci, lag.max = 10, type = "correlation", plot = TRUE,
    ylab="CCF", main = "Cross Correlation")
```



Northern Mockingbird Index x SLE Sentinel Chicken Seroconversions

```
resnmi<-resid(Arima(NMIndexPF78to13ts, model=fitnm))
resscnmi<-resid(Arima(log(SCSLEPF78to13ts+1), model=fitnm))
cat("\n")

ccf(resnmi, resscnmi, lag.max = 10, type = "correlation", plot = TRUE,
    ylab="CCF", main = "Cross Correlation")
```



Granger Causality

Blue Jay Average x SLE Sentinel Chicken Seroconversions

```
  grangertest(BJAveragePF78to15ts, log(SCSLEPF78to13ts+1), 8)

Granger causality test

Model 1: log(SCSLEPF78to13ts + 1) ~ Lags(log(SCSLEPF78to13ts + 1), 1:8) + Lags(BJAveragePF78to15ts, 1
:8)
Model 2: log(SCSLEPF78to13ts + 1) ~ Lags(log(SCSLEPF78to13ts + 1), 1:8)
```

```
  Res.Df Df      F Pr(>F)
1     11
2     19 -8 1.2515  0.356
```

Common Grackle Average x SLE Sentinel Chicken Seroconversions

```
  grangertest(log(SCSLEPF78to15ts+1), CGAveragePF78to15ts, 4)

Granger causality test

Model 1: CGAveragePF78to15ts ~ Lags(CGAveragePF78to15ts, 1:4) + Lags(log(SCSLEPF78to15ts + 1), 1:4)
Model 2: CGAveragePF78to15ts ~ Lags(CGAveragePF78to15ts, 1:4)
  Res.Df Df      F Pr(>F)
1     25
2     29 -4 1.0351 0.4088

  grangertest(log(SCSLEPF78to15ts+1), CGAveragePF78to15ts, 5)

Granger causality test

Model 1: CGAveragePF78to15ts ~ Lags(CGAveragePF78to15ts, 1:5) + Lags(log(SCSLEPF78to15ts + 1), 1:5)
Model 2: CGAveragePF78to15ts ~ Lags(CGAveragePF78to15ts, 1:5)
  Res.Df Df      F Pr(>F)
1     22
2     27 -5 0.9435 0.4727

  grangertest(CGAveragePF78to15ts, log(SCSLEPF78to13ts+1), 9)

Granger causality test

Model 1: log(SCSLEPF78to13ts + 1) ~ Lags(log(SCSLEPF78to13ts + 1), 1:9) + Lags(CGAveragePF78to15ts, 1
:9)
Model 2: log(SCSLEPF78to13ts + 1) ~ Lags(log(SCSLEPF78to13ts + 1), 1:9)
  Res.Df Df      F Pr(>F)
1      8
2     17 -9 2.0696 0.1595
```

Mourning Dove Average x SLE Sentinel Chicken Seroconversions

```
# No Cross Correlations to test
```

Northern Cardinal Average x SLE Sentinel Chicken Seroconversions

```
  grangertest(NCAveragePF78to15ts, log(SCSLEPF78to15ts+1), 2)

Granger causality test

Model 1: log(SCSLEPF78to15ts + 1) ~ Lags(log(SCSLEPF78to15ts + 1), 1:2) + Lags(NCAveragePF78to15ts, 1
:2)
Model 2: log(SCSLEPF78to15ts + 1) ~ Lags(log(SCSLEPF78to15ts + 1), 1:2)
  Res.Df Df      F Pr(>F)
1     31
2     33 -2 1.3774 0.2672

  grangertest(NCAveragePF78to15ts, log(SCSLEPF78to15ts+1), 8)

Granger causality test

Model 1: log(SCSLEPF78to15ts + 1) ~ Lags(log(SCSLEPF78to15ts + 1), 1:8) + Lags(NCAveragePF78to15ts, 1
:8)
Model 2: log(SCSLEPF78to15ts + 1) ~ Lags(log(SCSLEPF78to15ts + 1), 1:8)
  Res.Df Df      F Pr(>F)
1     13
2     21 -8 0.9963 0.4819

  grangertest(NCAveragePF78to15ts, log(SCSLEPF78to15ts+1), 9)
```

```
Granger causality test

Model 1: log(SCSLEPF78to15ts + 1) ~ Lags(log(SCSLEPF78to15ts + 1), 1:9) + Lags(NCAveragePF78to15ts, 1
:9)
Model 2: log(SCSLEPF78to15ts + 1) ~ Lags(log(SCSLEPF78to15ts + 1), 1:9)
  Res.Df Df      F Pr(>F)
1     10
2     19 -9 1.2534  0.363
```

Northern Mockingbird Average x SLE Sentinel Chicken Seroconversions

```
  grangertest(NMAveragePF78to15ts, log(SCSLEPF78to15ts+1), 8)

Granger causality test

Model 1: log(SCSLEPF78to15ts + 1) ~ Lags(log(SCSLEPF78to15ts + 1), 1:8) + Lags(NMAveragePF78to15ts, 1
:8)
Model 2: log(SCSLEPF78to15ts + 1) ~ Lags(log(SCSLEPF78to15ts + 1), 1:8)
  Res.Df Df      F Pr(>F)
1     13
2     21 -8 1.3004 0.3234

  grangertest(NMAveragePF78to15ts, log(SCSLEPF78to15ts+1), 9)

Granger causality test

Model 1: log(SCSLEPF78to15ts + 1) ~ Lags(log(SCSLEPF78to15ts + 1), 1:9) + Lags(NMAveragePF78to15ts, 1
:9)
Model 2: log(SCSLEPF78to15ts + 1) ~ Lags(log(SCSLEPF78to15ts + 1), 1:9)
  Res.Df Df      F Pr(>F)
1     10
2     19 -9 3.3928 0.0352 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Blue Jay Index x SLE Sentinel Chicken Seroconversions

```
# No Cross Correlations to test
```

Common Grackle Index x SLE Sentinel Chicken Seroconversions

```
  grangertest(CGIndexPF78to13ts, log(SCSLEPF78to13ts+1), 2)

Granger causality test

Model 1: log(SCSLEPF78to13ts + 1) ~ Lags(log(SCSLEPF78to13ts + 1), 1:2) + Lags(CGIndexPF78to13ts, 1:2
)
Model 2: log(SCSLEPF78to13ts + 1) ~ Lags(log(SCSLEPF78to13ts + 1), 1:2)
  Res.Df Df      F Pr(>F)
1     29
2     31 -2 2.3419 0.1141
```

Mourning Dove Index x SLE Sentinel Chicken Seroconversions

```
# No Cross Correlations to test
```

Northern Cardinal Index x SLE Sentinel Chicken Seroconversions

```
# No Cross Correlations to test
```

Northern Mockingbird Index x SLE Sentinel Chicken Seroconversions

```
# No Cross Correlations to test
```