Antarctic Sea Ice Time Series Project

Jason Rutberg

Import Data and Convert into Time Series Object

```
setwd("~/Desktop/Spring 2024/MA 585/Project")
Ice <- read.table("Sea Ice.txt")
Ice <- c(t(as.matrix(Ice)))
Ice <- ts(data = Ice, start = c(1989, 1), frequency = 12)
```

Plot Time Series (Figure 1)

```
plot(Ice, xlab = "Year",
     ylab = "Average Monthly Antarctic Sea Ice (Millions of Square Kilometers
)",
     main = "Figure 1: Average Monthly Antarctic Sea Ice from 1989 to 2023",
     col = "blue", cex.main = 2)
```

Classical Decomposition (Figure 2)

```
decomp.plot <- function(x, main = NULL, ...) {
if(is.null(main))
main <- paste("Decomposition of", x$type, "time series")
plot(cbind(observed = x$random + if (x$type == "additive")
x$trend + x$seasonal
else x$trend * x$seasonal, trend = x$trend, seasonal = x$seasonal, random = x
$random), main = main, ...)
}

z <- decompose(Ice, type = "additive")
decomp.plot(z, main = "Figure 2: Additive Decomposition of Sea Ice Data", cex
.main = 2)
```

ADF Test + Plot of Seasonally Differenced Data (Figure 3)

```
library(tseries)

adf.test(Ice) # Appears stationary (but we know it's NOT)

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

##
##  Augmented Dickey-Fuller Test
##
## data:  Ice
```

```
## Dickey-Fuller = -9.5004, Lag order = 7, p-value = 0.01
## alternative hypothesis: stationary
```

```
Ice_diff <- diff(diff(Ice, lag = 12)) # seasonal difference + first differenc
e
adf.test(Ice_diff) # stationary
```

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

```
##
##  Augmented Dickey-Fuller Test
##
## data:  Ice_diff
## Dickey-Fuller = -9.2703, Lag order = 7, p-value = 0.01
## alternative hypothesis: stationary
```

```
plot(Ice_diff, xlab = "Year",
     ylab = "Differenced Average Monthly Antarctic Sea Ice (Millions of Squar
e Kilometers)",
     main = "Figure 3: Differenced Average Monthly Antarctic Sea Ice",
     col = "orange", cex.main = 2)
```

ACF and PACF Plots (Figure 4)

```
ACF <- acf(Ice_diff, main = "Figure 4a: Differenced Data Sample ACF", ylim =
c(-1,1), lag.max = 50)

PACF <- pacf(Ice_diff, main = "Figure 4b: Differenced Data Sample PACF", ylim
= c(-1,1), lag.max = 50)
```

Manual SARIMA Model Selection

```
fit_1 <- Arima(Ice, order = c(1,1,1), seasonal = list(order = c(0,1,1), perio
d = 12))
fit_1
```

```
## Series: Ice
## ARIMA(1,1,1)(0,1,1)[12]
##
## Coefficients:
##          ar1      ma1     sma1
##       0.7126  -0.9024  -0.9999
## s.e.  0.0795   0.0532   0.2016
##
## sigma^2 = 0.08396:  log likelihood = -93.61
## AIC=195.22   AICc=195.32   BIC=211.26
```

```
fit_2 <- Arima(Ice, order = c(1,1,2), seasonal = list(order = c(0,1,1), perio
d = 12))
fit_2
```

```
## Series: Ice
## ARIMA(1,1,2)(0,1,1)[12]
##
## Coefficients:
##          ar1      ma1      ma2     sma1
##       0.4421  -0.5196  -0.2519  -0.9999
## s.e.  0.1089   0.1074   0.0576   0.2797
##
## sigma^2 = 0.08141:  log likelihood = -86.63
## AIC=183.26   AICc=183.41   BIC=203.31
```

```
fit_3 <- Arima(Ice, order = c(3,1,0), seasonal = list(order = c(0,1,1), perio
d = 12))
fit_3
```

```
## Series: Ice
## ARIMA(3,1,0)(0,1,1)[12]
##
## Coefficients:
##           ar1      ar2      ar3     sma1
##       -0.0366  -0.2372  -0.1280  -1.0000
## s.e.   0.0492   0.0478   0.0493   0.1429
##
## sigma^2 = 0.08393:  log likelihood = -92.66
## AIC=195.32   AICc=195.47   BIC=215.37
```

```
fit_4 <- Arima(Ice, order = c(0,1,3), seasonal = list(order = c(0,1,1), perio
d = 12))
fit_4
```

```
## Series: Ice
## ARIMA(0,1,3)(0,1,1)[12]
##
## Coefficients:
##           ma1      ma2      ma3     sma1
##       -0.0768  -0.3105  -0.1566  -1.0000
## s.e.   0.0492   0.0478   0.0471   0.1146
##
## sigma^2 = 0.08159:  log likelihood = -87.03
## AIC=184.06   AICc=184.21   BIC=204.1
```

```
fit_5 <- Arima(Ice, order = c(1,1,1), seasonal = list(order = c(0,1,2), perio
d = 12))
fit_5
```

```
## Series: Ice
## ARIMA(1,1,1)(0,1,2)[12]
##
## Coefficients:
##          ar1      ma1     sma1     sma2
##       0.7419  -0.9300  -0.8670  -0.1329
```

```
## s.e.  0.0933   0.0634   0.0688   0.0518
##
## sigma^2 = 0.08332:  log likelihood = -90.4
## AIC=190.79   AICc=190.94   BIC=210.84
```

```r
fit_6 <- Arima(Ice, order = c(1,1,2), seasonal = list(order = c(0,1,2), perio
d = 12))
fit_6
```

```
## Series: Ice
## ARIMA(1,1,2)(0,1,2)[12]
##
## Coefficients:
##          ar1      ma1      ma2     sma1     sma2
##       0.4547  -0.5403  -0.2438  -0.8931  -0.1068
## s.e.  0.1147   0.1139   0.0598   0.0693   0.0508
##
## sigma^2 = 0.08118:  log likelihood = -84.46
## AIC=180.92   AICc=181.13   BIC=204.97
```

```r
fit_7 <- Arima(Ice, order = c(3,1,0), seasonal = list(order = c(0,1,2), perio
d = 12))
fit_7
```

```
## Series: Ice
## ARIMA(3,1,0)(0,1,2)[12]
##
## Coefficients:
##           ar1      ar2      ar3     sma1     sma2
##       -0.0496  -0.2400  -0.1313  -0.8806  -0.1194
## s.e.   0.0497   0.0478   0.0494   0.0667   0.0507
##
## sigma^2 = 0.08353:  log likelihood = -89.96
## AIC=191.92   AICc=192.13   BIC=215.97
```

```r
fit_8 <- Arima(Ice, order = c(0,1,3), seasonal = list(order = c(0,1,2), perio
d = 12))
fit_8
```

```
## Series: Ice
## ARIMA(0,1,3)(0,1,2)[12]
##
## Coefficients:
##           ma1      ma2      ma3     sma1     sma2
##       -0.0852  -0.3061  -0.1543  -0.8947  -0.1053
## s.e.   0.0495   0.0486   0.0483   0.0676   0.0508
##
## sigma^2 = 0.0814:  log likelihood = -84.94
## AIC=181.87   AICc=182.08   BIC=205.92
```

```r
fit_9 <- Arima(Ice, order = c(1,1,1), seasonal = list(order = c(1,1,1), perio
d = 12))
fit_9

## Series: Ice
## ARIMA(1,1,1)(1,1,1)[12]
##
## Coefficients:
##          ar1      ma1    sar1     sma1
##       0.7479  -0.9333  0.1309  -1.0000
## s.e.  0.0982   0.0672  0.0534   0.0437
##
## sigma^2 = 0.08342:  log likelihood = -90.52
## AIC=191.04   AICc=191.19   BIC=211.09

fit_10 <- Arima(Ice, order = c(1,1,2), seasonal = list(order = c(1,1,1), peri
od = 12))
fit_10

## Series: Ice
## ARIMA(1,1,2)(1,1,1)[12]
##
## Coefficients:
##          ar1      ma1      ma2    sar1     sma1
##       0.4548  -0.5384  -0.2450  0.1056  -1.0000
## s.e.  0.1147   0.1137   0.0596  0.0514   0.0456
##
## sigma^2 = 0.08124:  log likelihood = -84.51
## AIC=181.02   AICc=181.23   BIC=205.08

fit_11 <- Arima(Ice, order = c(3,1,0), seasonal = list(order = c(1,1,1), peri
od = 12))
fit_11

## Series: Ice
## ARIMA(3,1,0)(1,1,1)[12]
##
## Coefficients:
##           ar1      ar2      ar3    sar1     sma1
##       -0.0473  -0.2400  -0.1293  0.1178  -1.0000
## s.e.   0.0495   0.0478   0.0493  0.0514   0.0419
##
## sigma^2 = 0.08362:  log likelihood = -90.03
## AIC=192.06   AICc=192.27   BIC=216.11

fit_12 <- Arima(Ice, order = c(0,1,3), seasonal = list(order = c(1,1,1), peri
od = 12))
fit_12

## Series: Ice
## ARIMA(0,1,3)(1,1,1)[12]
```

```
##
## Coefficients:
##           ma1      ma2      ma3    sar1     sma1
##       -0.0832  -0.3066  -0.1542  0.1030  -1.0000
## s.e.   0.0494   0.0486   0.0483  0.0513   0.0436
##
## sigma^2 = 0.08146:  log likelihood = -85.01
## AIC=182.02   AICc=182.23   BIC=206.07
```

SARIMA Model Diagnostics (Figure 5 and Figure A1)

```r
fit <- Arima(Ice, order = c(1,1,2), seasonal = list(order = c(0,1,2), period
= 12))

tsdiag(fit)
qqnorm(residuals(fit), main = "Figure A1: Normal Q-Q Plot of SARIMA Model Res
iduals", cex.main = 2)
qqline(residuals(fit))

auto.arima(residuals(fit), max.q = 0) # Order of Minimum AICC AR Model

## Series: residuals(fit)
## ARIMA(0,0,0) with zero mean
##
## sigma^2 = 0.0777:  log likelihood = -59.43
## AIC=120.86   AICc=120.87   BIC=124.9
```

SARIMA Model Forecast (Figure 6)

```r
ARIMAforecast <- forecast(fit, h = 24)
ARIMAforecast

##          Point Forecast      Lo 80     Hi 80       Lo 95     Hi 95
## Jan 2024      2.1850082  1.8145861  2.555430  1.61849648  2.751520
## Feb 2024      0.8176798  0.3158226  1.319537  0.05015538  1.585204
## Mar 2024      1.4551682  0.9014767  2.008860  0.60837013  2.301966
## Apr 2024      3.8007927  3.2165989  4.384986  2.90734534  4.694240
## May 2024      6.5847614  5.9778078  7.191715  5.65650603  7.513017
## Jun 2024      9.1733101  8.5469250  9.799695  8.21533683 10.131283
## Jul 2024     11.3411479 10.6969750 11.985321 10.35597051 12.326325
## Aug 2024     12.7361920 12.0751678 13.397216 11.72524271 13.747141
## Sep 2024     13.2064622 12.5292051 13.883719 12.17068679 14.242238
## Oct 2024     12.6581979 11.9651721 13.351224 11.59830633 13.718090
## Nov 2024     10.1824482  9.4740348 10.890862  9.09902342 11.265873
## Dec 2024      5.4992322  4.7757643  6.222700  4.39278355  6.605681
## Jan 2025      1.9214123  1.1716380  2.671187  0.77473142  3.068093
## Feb 2025      0.6336378 -0.1403631  1.407639 -0.55009446  1.817370
## Mar 2025      1.3379586  0.5437478  2.132169  0.12331803  2.552599
## Apr 2025      3.7295454  2.9170056  4.542085  2.48687305  4.972218
## May 2025      6.5647628  5.7348961  7.394629  5.29559129  7.833934
```

```
## Jun 2025      9.2199698  8.3733937 10.066546  7.92524337 10.514696
## Jul 2025     11.4006831 10.5378369 12.263529 10.08107372 12.720293
## Aug 2025     12.7752296 11.8964646 13.653995 11.43127457 14.119185
## Sep 2025     13.2118722 12.3174911 14.106253 11.84403439 14.579710
## Oct 2025     12.6577101 11.7479834 13.567437 11.26640322 14.049017
## Nov 2025     10.1462139  9.2213854 11.071042  8.73181070 11.560617
## Dec 2025      5.4240693  4.4843581  6.363781  3.98690506  6.861234
```

```
plot(ARIMAforecast, main = "Figure 6: Seasonal ARIMA (1,1,2) x (0,1,2)[12] Fo
recast", xlab = "Year", ylab = "Average Monthly Antarctic Sea Ice (Millions o
f Square Kilometers)", cex.main = 2)
```

Holt-Winters Forecast (Figure 7)

```
HWfit <- HoltWinters(Ice, seasonal = "additive")
HWforecast <- forecast(HWfit, h = 24)
HWforecast
```

```
##          Point Forecast        Lo 80     Hi 80      Lo 95     Hi 95
## Jan 2024       2.535739  2.112034795  2.959444  1.8877391  3.183740
## Feb 2024       1.347836  0.811863082  1.883809  0.5281362  2.167536
## Mar 2024       1.868988  1.240491431  2.497485  0.9077854  2.830191
## Apr 2024       4.185326  3.476278169  4.894374  3.1009308  5.269722
## May 2024       7.072052  6.290713122  7.853391  5.8770973  8.267007
## Jun 2024       9.813909  8.966423361 10.661394  8.5177917 11.110026
## Jul 2024      12.167384 11.258553291 13.076214 10.7774475 13.557320
## Aug 2024      13.620719 12.654429803 14.587007 12.1429074 15.098530
## Sep 2024      13.853213 12.832695525 14.873730 12.2924664 15.413959
## Oct 2024      13.142769 12.070763233 14.214775 11.5032777 14.782260
## Nov 2024      10.550340  9.429207984 11.671473  8.8357164 12.264964
## Dec 2024       5.730450  4.562255496  6.898645  3.9438506  7.517050
## Jan 2025       2.460862  1.228272975  3.693451  0.5757798  4.345944
## Feb 2025       1.272958 -0.002587317  2.548504 -0.6778204  3.223737
## Mar 2025       1.794110  0.477008441  3.111212 -0.2202233  3.808444
## Apr 2025       4.110449  2.753061866  5.467835  2.0345047  6.186393
## May 2025       6.997174  5.600664391  8.393684  4.8613967  9.132952
## Jun 2025       9.739031  8.304464615 11.173598  7.5450509 11.933011
## Jul 2025      12.092506 10.620866741 13.564146  9.8418279 14.343184
## Aug 2025      13.545841 12.038040013 15.053642 11.2398584 15.851824
## Sep 2025      13.778335 12.235219707 15.321450 11.4183438 16.138326
## Oct 2025      13.067891 11.490251988 14.645531 10.6551001 15.480683
## Nov 2025      10.475463  8.864038650 12.086887  8.0110023 12.939923
## Dec 2025       5.655573  4.011057879  7.300087  3.1405044  8.170641
```

```
plot(HWforecast, main = "Figure 7: Holt-Winters Forecast", xlab = "Year", yla
b = "Average Monthly Antarctic Sea Ice (Millions of Square Kilometers)", cex.
main = 2)
```

Model Performance

```r
train <- window(Ice, end = c(2019, 12))
test <- window(Ice, start = c(2020, 1)) # last 48 observations

# Holt-Winters Forecast
HWfit <- HoltWinters(train, seasonal = "additive")
HWforecast <- forecast(HWfit, h = 48)
HWforecast

##          Point Forecast        Lo 80     Hi 80       Lo 95     Hi 95
## Jan 2020       3.133112  2.718804114  3.547420  2.49948265  3.766742
## Feb 2020       1.775124  1.259169359  2.291078  0.98603967  2.564208
## Mar 2020       2.288781  1.688142452  2.889419  1.37018373  3.207378
## Apr 2020       4.730717  4.055939771  5.405494  3.69873441  5.762699
## May 2020       7.686917  6.945376892  8.428457  6.55282933  8.821004
## Jun 2020      10.582066  9.779296522 11.384836  9.35433592 11.809797
## Jul 2020      12.767260 11.907610905 13.626909 11.45254010 14.081980
## Aug 2020      13.994703 13.081711323 14.907695 12.59840256 15.391004
## Sep 2020      14.253816 13.290429922 15.217201 12.78044435 15.727187
## Oct 2020      13.724182 12.712910982 14.735453 12.17757632 15.270788
## Nov 2020      11.062213 10.005223303 12.119203  9.44568678 12.678739
## Dec 2020       6.523430  5.422619510  7.624241  4.83988540  8.206976
## Jan 2021       3.058235  1.896246510  4.220223  1.28112717  4.835342
## Feb 2021       1.700246  0.498258516  2.902233 -0.13803522  3.538527
## Mar 2021       2.213903  0.973205162  3.454601  0.31641932  4.111387
## Apr 2021       4.655839  3.377602425  5.934076  2.70094478  6.610733
## May 2021       7.612039  6.297335244  8.926743  5.60137297  9.622706
## Jun 2021      10.507189  9.157001831 11.857375  8.44225614 12.572121
## Jul 2021      12.692382 11.307621909 14.077143 10.57457394 14.810191
## Aug 2021      13.919826 12.501333713 15.338318 11.75042942 16.089222
## Sep 2021      14.178938 12.727498321 15.630378 11.95915258 16.398723
## Oct 2021      13.649305 12.165648771 15.132960 11.38024881 15.918360
## Nov 2021      10.987335  9.472148140 12.502523  8.67005649 13.304614
## Dec 2021       6.448553  4.902477205  7.994628  4.08403421  8.813071
## Jan 2022       2.983357  1.393142705  4.573571  0.55133412  5.415380
## Feb 2022       1.625368  0.005695873  3.245041 -0.85170696  4.102444
## Mar 2022       2.139026  0.490421086  3.787630 -0.38229739  4.660348
## Apr 2022       4.580962  2.903924187  6.257999  2.01615425  7.145769
## May 2022       7.537162  5.832165524  9.242158  4.92959509 10.144728
## Jun 2022      10.432311  8.699807174 12.164815  7.78267507 13.081947
## Jul 2022      12.617505 10.857923391 14.377086  9.92645722 15.308553
## Aug 2022      13.844948 12.058699316 15.631197 11.11311635 16.576780
## Sep 2022      14.104060 12.291536648 15.916584 11.33204456 16.876076
## Oct 2022      13.574427 11.736003817 15.412850 10.76280138 16.386053
## Nov 2022      10.912458  9.048494907 12.776421  8.06177260 13.763143
## Dec 2022       6.373675  4.484518002  8.262832  3.48445859  9.262892
## Jan 2023       2.908479  0.983032253  4.833926 -0.03623787  5.853197
## Feb 2023       1.550491 -0.399356452  3.500338 -1.43154321  4.532525
## Mar 2023       2.064148  0.090202183  4.038094 -0.95474156  5.083037
```

```
## Apr 2023       4.506084  2.508330327  6.503837  1.45078347  7.561384
## May 2023       7.462284  5.441003024  9.483565  4.37100149 10.553567
## Jun 2023      10.357433  8.312895621 12.401971  7.23058273 13.484284
## Jul 2023      12.542627 10.475094397 14.610160  9.38060862 15.704646
## Aug 2023      13.770070 11.679795411 15.860345 10.57327065 16.966870
## Sep 2023      14.029183 11.916410312 16.141955 10.79797615 17.260389
## Oct 2023      13.499549 11.364516611 15.634582 10.23429854 16.764800
## Nov 2023      10.837580  8.680516705 12.994643  7.53863632 14.136524
## Dec 2023       6.298798  4.119926305  8.477669  2.96650152  9.631094
```

```
HWerr <- test - HWforecast$mean
HWrmse <- sqrt(mean(HWerr^2))
HWmae <- mean(abs(HWerr))
HWmape <- mean(abs((HWerr*100)/test))

HWrmse # Holt-Winters RMSE
```

```
## [1] 0.7132303
```

```
HWmae # Holt-Winters MAE
```

```
## [1] 0.5478606
```

```
HWmape # Holt-Winters MAPE
```

```
## [1] 9.405666
```

```
# SARIMA(1,1,2)x(0,1,2)[12] Forecast
arimafit <- Arima(train, order = c(1,1,2), seasonal = list(order = c(0,1,2),
period = 12))
arimafcast <- forecast(arimafit, h = 48)
arimafcast
```

```
##          Point Forecast      Lo 80     Hi 80       Lo 95     Hi 95
## Jan 2020       3.025450  2.6585098  3.392391  2.46426320  3.586637
## Feb 2020       1.715924  1.2299230  2.201926  0.97264932  2.459200
## Mar 2020       2.360701  1.8301858  2.891217  1.54934779  3.172055
## Apr 2020       4.729566  4.1725171  5.286616  3.87763310  5.581500
## May 2020       7.547356  6.9697964  8.124915  6.66405496  8.430657
## Jun 2020      10.239618  9.6440029 10.835234  9.32870300 11.150534
## Jul 2020      12.459174 11.8466994 13.071649 11.52247487 13.395873
## Aug 2020      13.831110 13.2024860 14.459734 12.86971243 14.792508
## Sep 2020      14.290088 13.6458177 14.934357 13.30476185 15.275413
## Oct 2020      13.733836 13.0743258 14.393346 12.72520242 14.742469
## Nov 2020      11.216815 10.5424138 11.891215 10.18540764 12.248222
## Dec 2020       6.551194  5.8622146  7.240173  5.49749099  7.604897
## Jan 2021       2.959817  2.2455474  3.674086  1.86743601  4.052198
## Feb 2021       1.660487  0.9234809  2.397494  0.53333321  2.787642
## Mar 2021       2.366973  1.6110535  3.122891  1.21089422  3.523051
## Apr 2021       4.770759  3.9975653  5.543952  3.58826155  5.953255
## May 2021       7.610913  6.8212779  8.400548  6.40327040  8.818555
## Jun 2021      10.285224  9.4796616 11.090787  9.05322240 11.517226
```

```
## Jul 2021      12.467095 11.6459833 13.288207 11.21131299 13.722877
## Aug 2021      13.810145 12.9737999 14.646491 12.53106543 15.089225
## Sep 2021      14.250438 13.3991391 15.101736 12.94848885 15.552386
## Oct 2021      13.713014 12.8470163 14.579011 12.38858507 15.037442
## Nov 2021      11.209059 10.3285925 12.089526  9.86250163 12.555617
## Dec 2021       6.531375  5.6366408  7.426110  5.16299693  7.899754
## Jan 2022       2.944009  2.0328733  3.855144  1.55054740  4.337470
## Feb 2022       1.650218  0.7234528  2.576982  0.23285312  3.067582
## Mar 2022       2.358942  1.4174530  3.300430  0.91905902  3.798824
## Apr 2022       4.763633  3.8079006  5.719365  3.30196663  6.225299
## May 2022       7.604153  6.6344868  8.573819  6.12117664  9.087129
## Jun 2022      10.278612  9.2952495 11.261975  8.77468859 11.782536
## Jul 2022      12.460543 11.4636866 13.457399 10.93598268 13.985103
## Aug 2022      13.803617 12.7934522 14.813782 12.25870319 15.348531
## Sep 2022      14.243919 13.2206174 15.267221 12.67891411 15.808925
## Oct 2022      13.706499 12.6702183 14.742780 12.12164429 15.291354
## Nov 2022      11.202546 10.1534254 12.251667  9.59805434 12.807039
## Dec 2022       6.524863  5.4630125  7.586714  4.90090267  8.148824
## Jan 2023       2.937497  1.8610040  4.013990  1.29114309  4.583850
## Feb 2023       1.643706  0.5532442  2.734167 -0.02401125  3.311423
## Mar 2023       2.352430  1.2487442  3.456116  0.66448826  4.040371
## Apr 2023       4.757121  3.6405914  5.873651  3.04953638  6.464706
## May 2023       7.597641  6.4685026  8.726780  5.87077270  9.324510
## Jun 2023      10.272101  9.1305275 11.413674  8.52621512 12.017986
## Jul 2023      12.454031 11.3001708 13.607892 10.68935403 14.218708
## Aug 2023      13.797105 12.6310911 14.963120 12.01384040 15.580370
## Sep 2023      14.237408 13.0593625 15.415453 12.43574309 16.039072
## Oct 2023      13.699988 12.5100233 14.889952 11.88009428 15.519881
## Nov 2023      11.196035  9.9942446 12.397825  9.35805539 13.034014
## Dec 2023       6.518352  5.3047964  7.731907  4.66237919  8.374324
```

```r
arimaerr <- test - arimafcast$mean
arimarmse <- sqrt(mean(arimaerr^2))
arimamae <- mean(abs(arimaerr))
arimamape <- mean(abs((arimaerr*100)/test))

arimarmse # ARIMA RMSE

## [1] 0.7635105

arimamae # ARIMA MAE

## [1] 0.6142785

arimamape # ARIMA MAPE

## [1] 10.80418

Measure <- c("SARIMA(1,1,2)x(0,1,2)[12]", "Holt-Winters")
RMSE <- round(c(arimarmse, HWrmse), 4)
MAE <- round(c(arimamae, HWmae), 4)
```

```r
MAPE <- round(c(arimamape, HWmape), 4)
rbind(Measure, RMSE, MAE, MAPE)
```

```
##         [,1]                          [,2]
## Measure "SARIMA(1,1,2)x(0,1,2)[12]" "Holt-Winters"
## RMSE    "0.7635"                      "0.7132"
## MAE     "0.6143"                      "0.5479"
## MAPE    "10.8042"                     "9.4057"
```