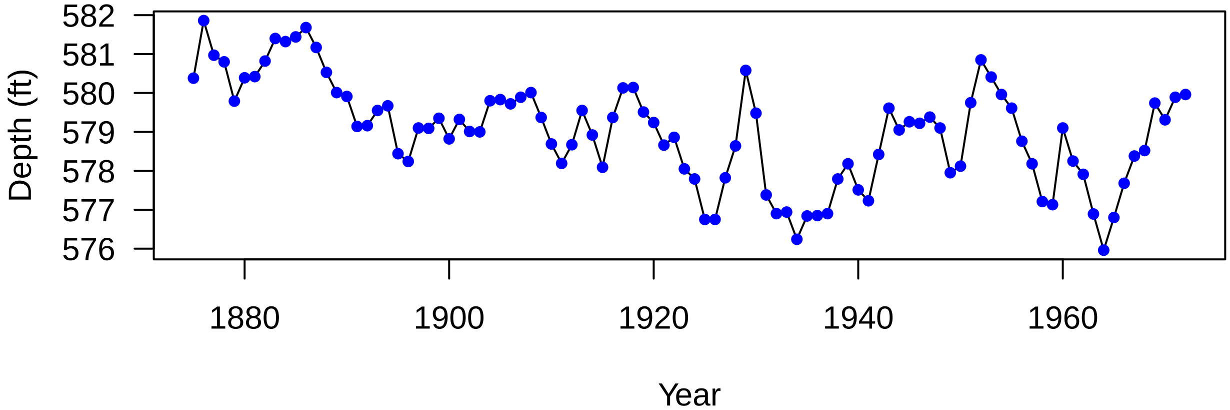
MATH 4070: ARMA Models: Estimation, Diagnostics, and Model Selection

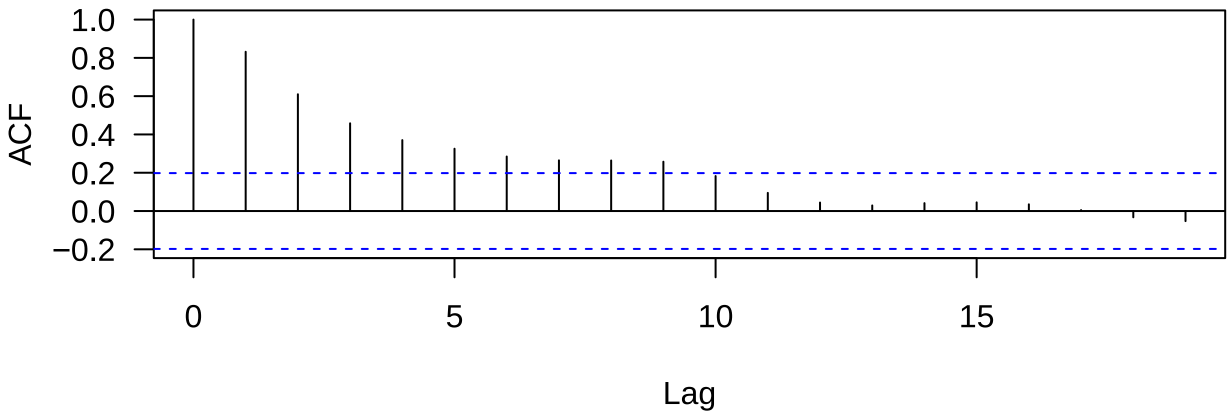
Whitney Huang, Clemson University

Contents

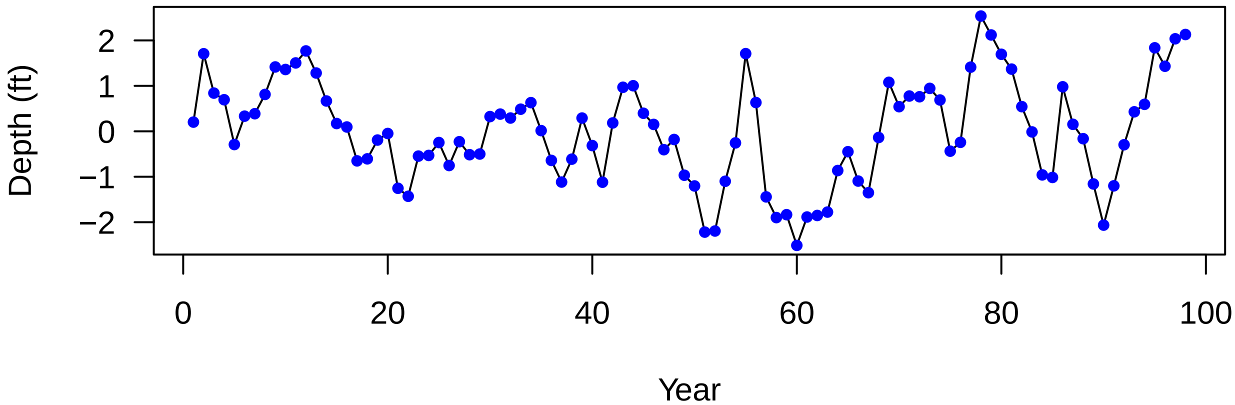
Lake Huron Example

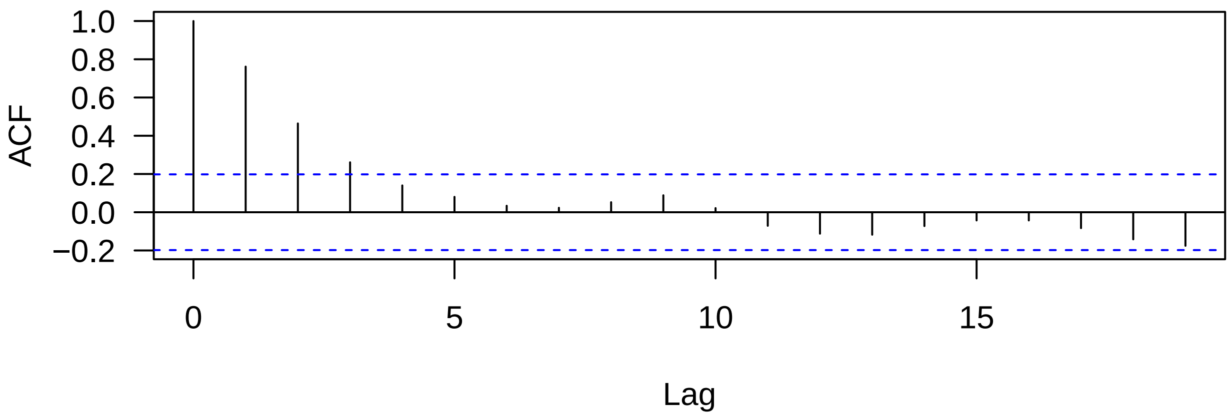
```r
data(LakeHuron)
par(las = 1, mfrow = c(2, 1), mar = c(4, 4, 0.8, 0.6))
plot(LakeHuron, ylab = "Depth (ft)", xlab = "Year")
points(LakeHuron, cex = 0.8, col = "blue", pch = 16)
acf(LakeHuron)
```



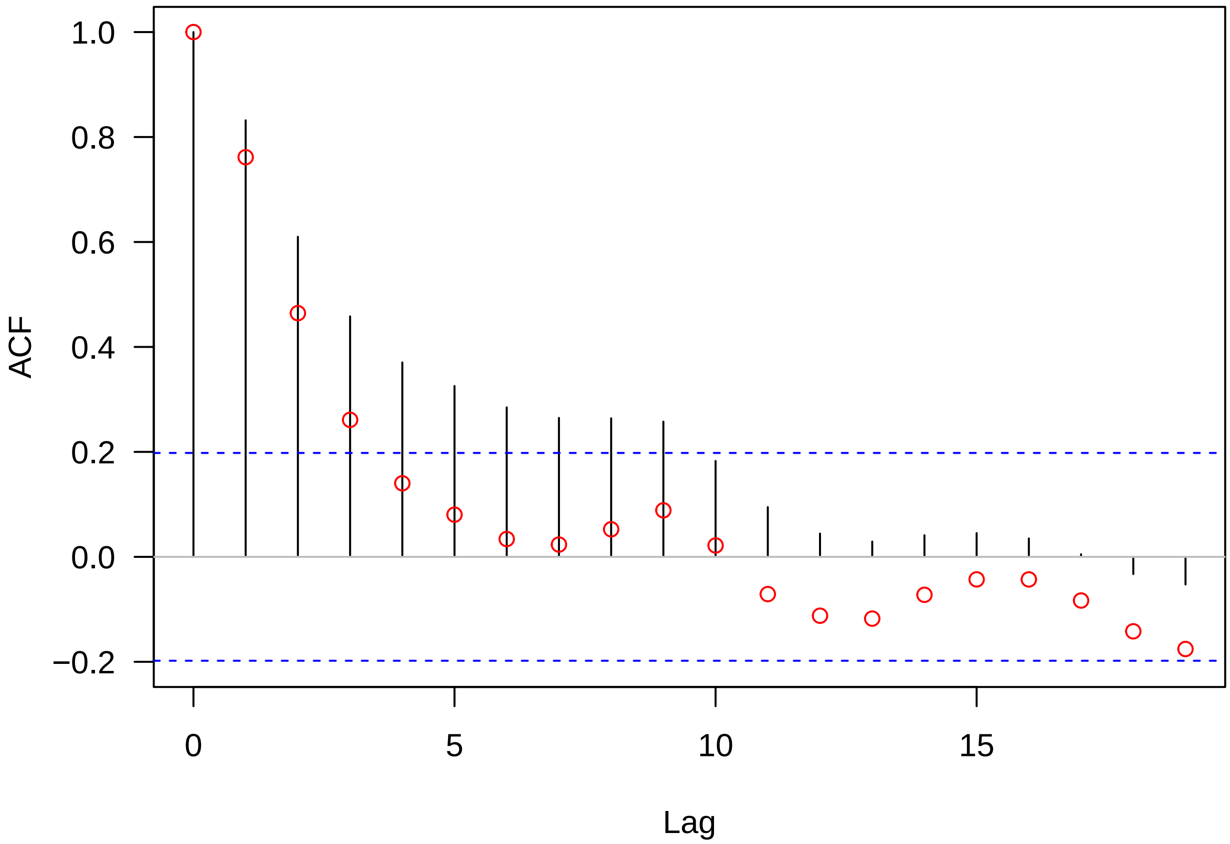



```
# Let's remove the (linear trend)
yr <- 1875:1972
lm <- lm(LakeHuron ~ yr)
plot(lm$residuals, ylab = "Depth (ft)", xlab = "Year", type = "l")
points(lm$residuals, cex = 0.8, col = "blue", pch = 16)
stats::acf(lm$residuals)
```

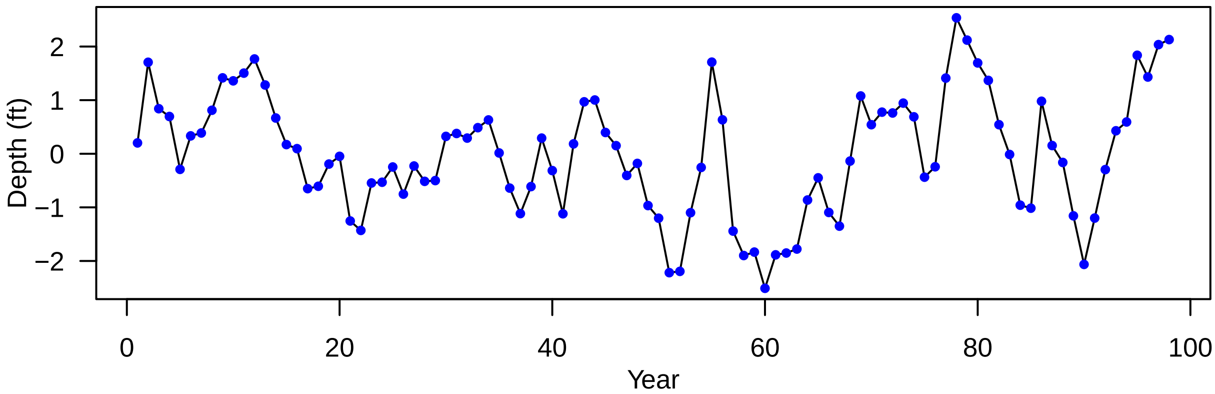


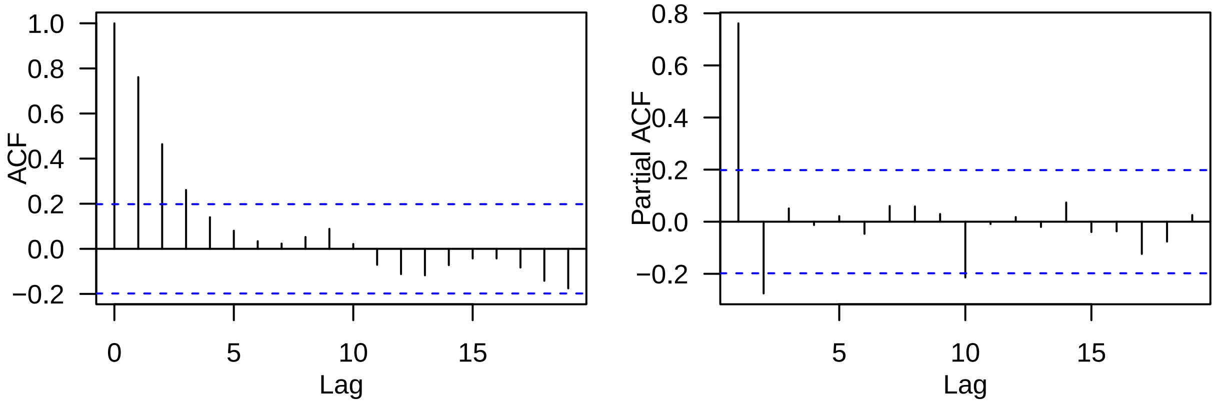



```
par(mfrow = c(1, 1), las = 1)
plot(0:19, stats::acf(LakeHuron, plot = F)$acf, type = "h", xlab = "Lag", ylab = "ACF",
     ylim = c(-0.2, 1))
abline(h = 0, col = "gray")
abline(h = c(-1, 1) * qnorm(0.975) / sqrt(length(LakeHuron)) , col = "blue", lty = 2)
acf_detrend <- stats::acf(lm$residuals, plot = F)$acf
points(0:19, acf_detrend, col = "red")
```




```
par(las = 1, mgp = c(2, 1, 0), mar = c(3.6, 3.6, 0.8, 0.6))
layout(matrix(c(1, 1, 2, 3), 2, 2, byrow = TRUE))
plot(lm$residuals, ylab = "Depth (ft)", xlab = "Year", type = "l")
points(lm$residuals, cex = 0.8, col = "blue", pch = 16)
acf(lm$residuals)
pacf(lm$residuals)
```





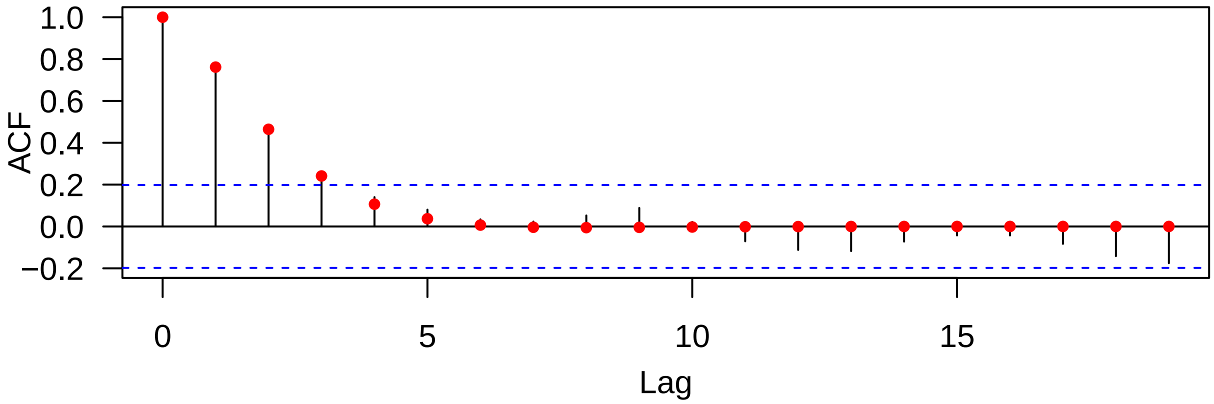
Yule-Walker Estimate

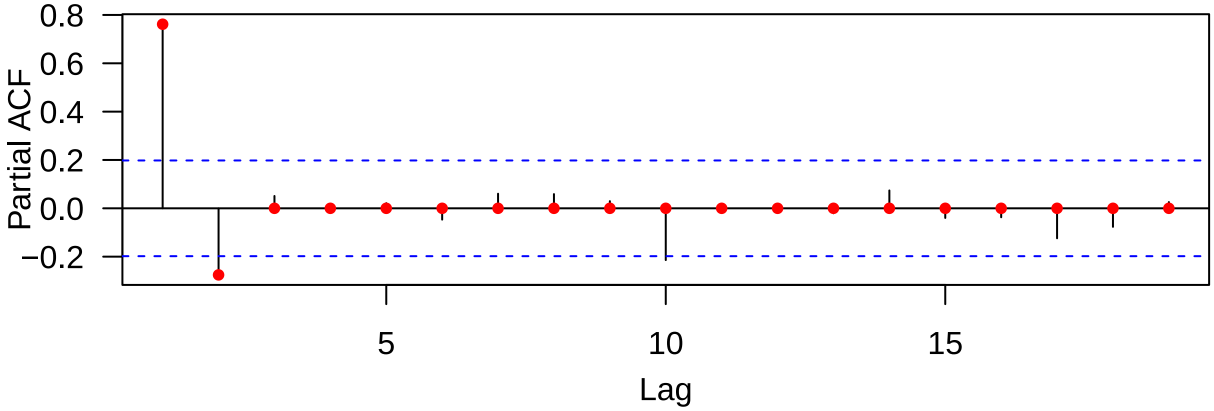
Let's fit an AR(2) model using the Yule-Walker method on the detrended Lake Huron series

```
YW_est <- ar(lm$residuals, aic = F, order.max = 2, method = "yw")
YW_est
```

```
##
## Call:
## ar(x = lm$residuals, aic = F, order.max = 2, method = "yw")
##
## Coefficients:
##       1        2
##  0.9714  -0.2754
##
## Order selected 2  sigma^2 estimated as  0.501
```

```
# plot sample and estimated acf/pacf
par(las = 1, mgp = c(2.2, 1, 0), mar = c(3.6, 3.6, 0.6, 0.6), mfrow = c(2, 1))
acf(lm$residuals)
acf_YWest <- ARMAacf(ar = YW_est$ar, lag.max = 23)
points(0:23, acf_YWest, col = "red", pch = 16, cex = 0.8)
pacf(lm$residuals)
pacf_YWest <- ARMAacf(ar = YW_est$ar, lag.max = 23, pacf = T)
points(1:23, pacf_YWest, col = "red", pch = 16, cex = 0.8)
```





MLE

Here, we fit an AR(2) model using the method of maximum likelihood on the detrended Lake Huron series

```
(MLE_est1 <- arima(lm$residuals, order = c(2, 0, 0), method = "ML"))
```

```
##
## Call:
## arima(x = lm$residuals, order = c(2, 0, 0), method = "ML")
##
## Coefficients:
##          ar1      ar2  intercept
##       1.0047  -0.2919     0.0197
## s.e.  0.0977   0.1004     0.2350
##
## sigma^2 estimated as 0.4571:  log likelihood = -101.25,  aic = 210.5
```

Below, try a few other ARMA models and set the mean equal to zero

```
(MLE_est2 <- arima(lm$residuals, order = c(1, 0, 0), include.mean = F))
```

```
##
## Call:
## arima(x = lm$residuals, order = c(1, 0, 0), include.mean = F)
##
```

```
## Coefficients:
##          ar1
##       0.7826
## s.e.  0.0635
##
## sigma^2 estimated as 0.4975:  log likelihood = -105.32,  aic = 214.65
```

```
(MLE_est3 <- arima(lm$residuals, order = c(2, 0, 1), include.mean = F))
```

```
##
## Call:
## arima(x = lm$residuals, order = c(2, 0, 1), include.mean = F)
##
## Coefficients:
##          ar1      ar2     ma1
##       0.8381  -0.1631  0.1842
## s.e.  0.3178   0.2618  0.3179
##
## sigma^2 estimated as 0.4556:  log likelihood = -101.09,  aic = 210.19
```

Use MLE to fit the trend and ARMA model in one step

```
(MLE_est4 <- arima(LakeHuron, order = c(2, 0, 0), xreg = yr))
```

```
##
## Call:
## arima(x = LakeHuron, order = c(2, 0, 0), xreg = yr)
##
## Coefficients:
##          ar1      ar2  intercept       yr
##       1.0048  -0.2913   620.5115  -0.0216
## s.e.  0.0976   0.1004    15.5771   0.0081
##
## sigma^2 estimated as 0.4566:  log likelihood = -101.2,  aic = 212.4
```

```
library(forecast)
```

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

```
MLE_est4 <- Arima(LakeHuron, order = c(2, 0, 0), xreg = yr)
```

Model selection

```
orders <- list(
  c(1, 0, 0),  # ARMA(1,0)
  c(1, 0, 1),  # ARMA(2,1)
  c(2, 0, 0),  # ARMA(1,2)
```

```r
  c(2, 0, 1)   # ARMA(2,2)
  )
models <- c("ARMA(1,0)", "ARMA(1,1)", "ARMA(2,0)", "ARMA(2,1)")

fit <- lapply(orders, function(z) arima(LakeHuron, order = z, xreg = yr))
names(fit) <- models

lapply(fit, AIC)
```

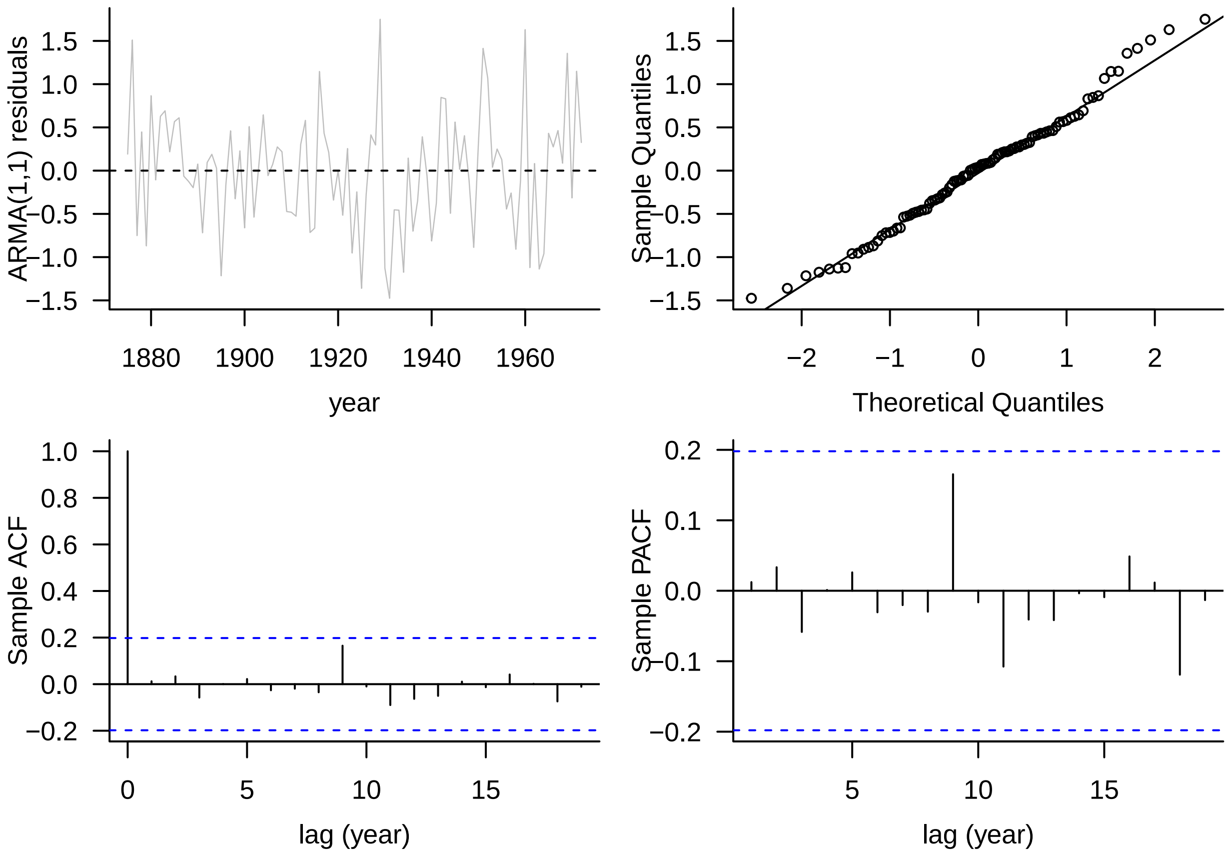
```
## $`ARMA(1,0)`
## [1] 218.4501
##
## $`ARMA(1,1)`
## [1] 212.3954
##
## $`ARMA(2,0)`
## [1] 212.3965
##
## $`ARMA(2,1)`
## [1] 214.0638
```

```r
library(MuMIn)
lapply(fit, AICc)
```

```
## $`ARMA(1,0)`
## [1] 218.8803
##
## $`ARMA(1,1)`
## [1] 213.0476
##
## $`ARMA(2,0)`
## [1] 213.0487
##
## $`ARMA(2,1)`
## [1] 214.9868
```

Model Diagnostics

```r
## extract the residuals
resids <- resid(fit[[2]])
## time series plot of the residuals
par(bty = "L", mar = c(3.6, 3.6, 0.5, 0.6), las = 1, mgp = c(2.4, 1, 0),
    mfrow = c(2, 2))
plot(yr, resids, type = "l", xlab = "year",
     ylab = "ARMA(1,1) residuals", lwd = 0.6, col = "gray")
abline(h = 0, lty = 2)
## Normal Q-Q plot for the residuals
qqnorm(resids, main = "", cex = 0.75); qqline(resids)
## Sample ACF and PACF of the residuals
acf(resids, ylab = "Sample ACF", xlab = "lag (year)", main = "")
pacf(resids, ylab = "Sample PACF", xlab = "lag (year)")
```




```r
## Carry out the Ljung-Box test
Box.test(resids, lag = 10, type = "Ljung-Box")
```

```
##
##  Box-Ljung test
##
## data:  resids
## X-squared = 3.7882, df = 10, p-value = 0.9564
```

Monte Carlo Simulation

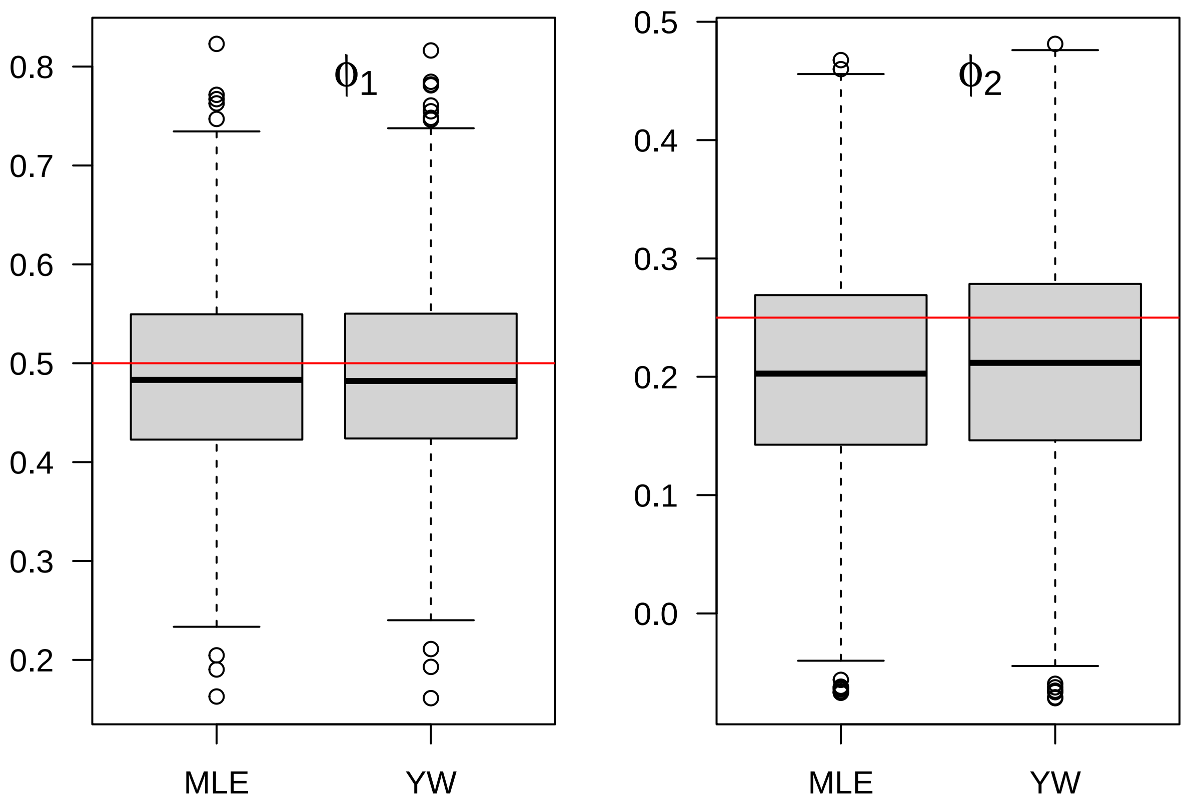
```r
N = 1000
n = 100
phi <- c(0.5, 0.25)
sim1 <- replicate(N, arima.sim(n = n, list(ar = phi)))
```

```r
yw <- apply(sim1, 2, ar, aic = F, order.max = 2, method = "yule-walker")
mle <- apply(sim1, 2, ar, aic = F, order.max = 2, method = "mle")
```

```r
yw_phi <- t(array(unlist(lapply(yw, function(x) x$ar)), dim = c(2, N)))
mle_phi <- t(array(unlist(lapply(mle, function(x) x$ar)), dim = c(2, N)))
```

```r
par(mar = c(3.6, 3.6, 0.5, 0.6), las = 1, mgp = c(2.2, 1, 0), mfrow = c(1, 2))
boxplot(yw_phi[, 1], mle_phi[, 1], xaxt = "n")
abline(h = 0.5, col = "red")
axis(1, at = 1:2, labels = c("MLE", "YW"))
legend("top", legend = expression(phi[1]), bty = "n", cex = 1.5)

boxplot(yw_phi[, 2], mle_phi[, 2], xaxt = "n")
abline(h = 0.25, col = "red")
axis(1, at = 1:2, labels = c("MLE", "YW"))
legend("top", legend = expression(phi[2]), bty = "n", cex = 1.5)
```




```r
apply(yw_phi, 2, mean); apply(yw_phi, 2, sd)
```

```
## [1] 0.4866833 0.2035050
```

```
## [1] 0.09558568 0.09231376
```

```r
apply(mle_phi, 2, mean); apply(mle_phi, 2, sd)
```

```
## [1] 0.4880688 0.2116262
```

```
## [1] 0.09674300 0.09571005
```

```r
sqrt(mean((yw_phi[, 1] - 0.5)^2))
```

```
## [1] 0.09646149
```

```r
sqrt(mean((mle_phi[, 1] - 0.5)^2))
```

```
## [1] 0.09742793
```

```r
roots <- t(apply(yw_phi, 1, function(x) Mod(polyroot(c(1, -x[1], -x[2])))))
check <- apply(roots, 1, function(x) ifelse(x[1] > 1 && x[2] > 1, 0, 1))
```