

2021_MARSSModel

##Read in data

```
KSV_meantemps <- readRDS('KSV_meantemps.rds')
daily_means_long_klamath <- readRDS('daily_means_long.rds')
covariate_klamath <- readRDS('covariate.rds')

daily_means_long_klamath <- rbind(daily_means_long_klamath, KSV = KSV_meantemps)
str(daily_means_long_klamath)
```

##Data Matrix

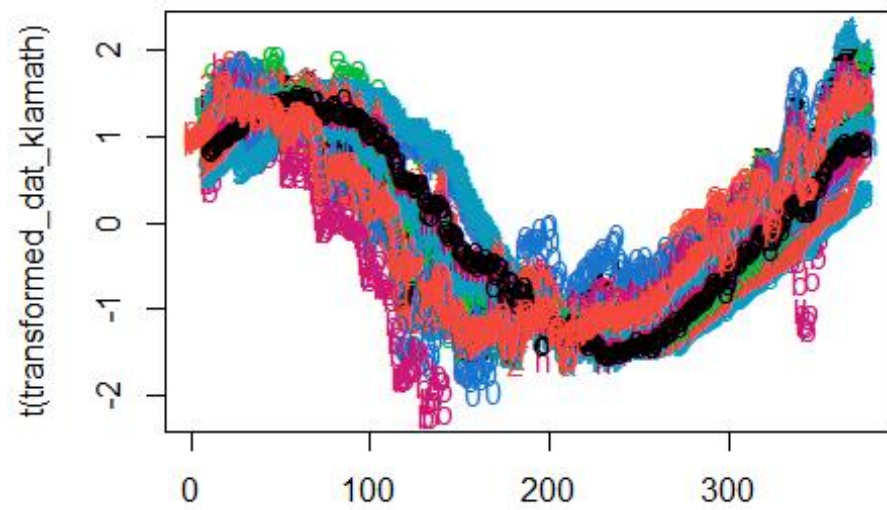
```
#Convert data to matrix and z-score
transformed_dat_klamath <- as.matrix(daily_means_long_klamath)
transformed_dat_klamath <- zscore(transformed_dat_klamath)
saveRDS(transformed_dat_klamath, "transformed_dat_klamath.rds")
```

##Covariates

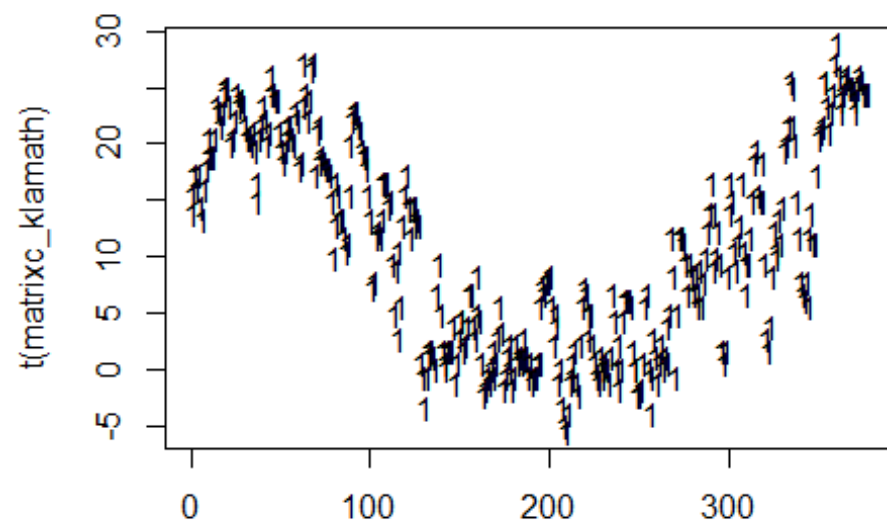
```
#Build the little c matrix, call it matrixc
matrixc_klamath <- matrix(nrow=1, ncol=378)
matrixc_klamath <- (as.matrix(covariate_klamath))
saveRDS(matrixc_klamath, "matrixc_klamath.rds")
```

###Check data and covariates

```
matplot(t(transformed_dat_klamath))
```



```
matplot(t(matrixxc_klamath))
```



#Looks okay, we removed AP2 in the visualization .rmd file

```
##Z-Matrices ###matrix2_klamath
```

```
#Hypothesis 1: ALL ponds and creeks and mainstem are separate
```

```
matrix2_klamath <- matrix(nrow=26,ncol=12)
matrix2_klamath[c(1:2),1] <- 1 #Alexander
matrix2_klamath[c(1:2),c(2:12)] <- 0
matrix2_klamath[c(4:5),2] <- 1 #Stender
matrix2_klamath[c(4:5),c(1,3:12)] <- 0
matrix2_klamath[c(3,6,9,11,12),3] <- 1 #Seiad Creek
matrix2_klamath[c(3,6,9,11,12),c(1,2,4:12)] <- 0
matrix2_klamath[7,4] <- 1 #Durazo
matrix2_klamath[7,c(1:3,5:12)] <- 0
matrix2_klamath[8,5] <- 1 #Lower Seiad
matrix2_klamath[8,c(1:4,6:12)] <- 0
matrix2_klamath[10,6] <- 1 #May
matrix2_klamath[10,c(1:5,7:12)] <- 0
matrix2_klamath[c(12:13),7] <- 1 #Fish Gulch
matrix2_klamath[c(12:13),c(1:6,8:12)] <- 0
matrix2_klamath[c(15:17),8] <- 1 #Goodman
matrix2_klamath[c(15:17),c(1:7,9:12)] <- 0
matrix2_klamath[c(19:21),9] <- 1 #Upper Lawrence
matrix2_klamath[c(19:21),c(1:8,10:12)] <- 0
matrix2_klamath[c(23:25),10] <- 1 #Lower Lawrence
matrix2_klamath[c(23:25),c(1:9,11:12)] <- 0
matrix2_klamath[c(14,18,22),11] <- 1 #Horse Creek
matrix2_klamath[c(14,18,22),c(1:10,12)] <- 0
matrix2_klamath[26,12] <- 1 #Klamath
matrix2_klamath[26,c(1:11)] <- 0
matrix2_klamath
```

```
##      [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10] [,11] [,12]
## [1,]    1    0    0    0    0    0    0    0    0    0    0    0
## [2,]    1    0    0    0    0    0    0    0    0    0    0    0
## [3,]    0    0    1    0    0    0    0    0    0    0    0    0
## [4,]    0    1    0    0    0    0    0    0    0    0    0    0
## [5,]    0    1    0    0    0    0    0    0    0    0    0    0
## [6,]    0    0    1    0    0    0    0    0    0    0    0    0
## [7,]    0    0    0    1    0    0    0    0    0    0    0    0
## [8,]    0    0    0    0    1    0    0    0    0    0    0    0
## [9,]    0    0    1    0    0    0    0    0    0    0    0    0
## [10,]   0    0    0    0    0    1    0    0    0    0    0    0
## [11,]   0    0    1    0    0    0    0    0    0    0    0    0
## [12,]   0    0    0    0    0    0    1    0    0    0    0    0
## [13,]   0    0    0    0    0    0    1    0    0    0    0    0
## [14,]   0    0    0    0    0    0    0    0    0    0    1    0
## [15,]   0    0    0    0    0    0    0    1    0    0    0    0
## [16,]   0    0    0    0    0    0    0    1    0    0    0    0
## [17,]   0    0    0    0    0    0    0    1    0    0    0    0
## [18,]   0    0    0    0    0    0    0    0    0    0    1    0
## [19,]   0    0    0    0    0    0    0    0    1    0    0    0
```

```
## [20,] 0 0 0 0 0 0 0 0 0 1 0 0 0
## [21,] 0 0 0 0 0 0 0 0 0 1 0 0 0
## [22,] 0 0 0 0 0 0 0 0 0 0 0 1 0
## [23,] 0 0 0 0 0 0 0 0 0 0 1 0 0
## [24,] 0 0 0 0 0 0 0 0 0 0 1 0 0
## [25,] 0 0 0 0 0 0 0 0 0 0 1 0 0
## [26,] 0 0 0 0 0 0 0 0 0 0 0 0 1
```

###matrix3_klamath

#Hypothesis 2: ponds versus creeks versus Klamath

```
matrix3_klamath <- matrix(nrow=26, ncol=3)
matrix3_klamath[c(1:2,4:5,7:8,10,12:13,15:17,19:21,23:25),1] <- 1 #ALL ponds
matrix3_klamath[c(1:2,4:5,7:8,10,12:13,15:17,19:21,23:25),c(2:3)] <- 0
matrix3_klamath[c(3,6,9,11,14,18,22),c(1,3)] <- 0 #ALL creeks
matrix3_klamath[c(3,6,9,11,14,18,22),2] <- 1
matrix3_klamath[26,3] <- 1 #Klamath
matrix3_klamath[26,c(1:2)] <- 0
matrix3_klamath
```

```
##      [,1] [,2] [,3]
## [1,] 1 0 0
## [2,] 1 0 0
## [3,] 0 1 0
## [4,] 1 0 0
## [5,] 1 0 0
## [6,] 0 1 0
## [7,] 1 0 0
## [8,] 1 0 0
## [9,] 0 1 0
## [10,] 1 0 0
## [11,] 0 1 0
## [12,] 1 0 0
## [13,] 1 0 0
## [14,] 0 1 0
## [15,] 1 0 0
## [16,] 1 0 0
## [17,] 1 0 0
## [18,] 0 1 0
## [19,] 1 0 0
## [20,] 1 0 0
## [21,] 1 0 0
## [22,] 0 1 0
## [23,] 1 0 0
## [24,] 1 0 0
## [25,] 1 0 0
## [26,] 0 0 1
```

###matrix4_klamath

#Hypothesis 3: tributary versus tributary versus Klamath

```
matrix4_klamath <-matrix(nrow=26,ncol=5)
matrix4_klamath[c(1:2,4:5,7:8,10),1] <- 1 #Seiad Creek Ponds
matrix4_klamath[c(1:2,4:5,7:8,10),c(2:5)] <- 0
matrix4_klamath[c(3,6,9,11),2] <- 1 #Seiad Creek
matrix4_klamath[c(3,6,9,11),c(1,3:5)] <- 0
matrix4_klamath[c(12:13,15:17,19:21,23:25),3] <- 1 #Horse Creek Ponds
matrix4_klamath[c(12:13,15:17,19:21,23:25),c(1,2,4,5)] <- 0
matrix4_klamath[c(14,18,22),4] <- 1 #Horse Creek
matrix4_klamath[c(14,18,22),c(1:3,5)] <- 0
matrix4_klamath[26,5] <- 1 #Klamath
matrix4_klamath[c(26),c(1:4)] <- 0
matrix4_klamath
```

```
##      [,1] [,2] [,3] [,4] [,5]
## [1,]    1    0    0    0    0
## [2,]    1    0    0    0    0
## [3,]    0    1    0    0    0
## [4,]    1    0    0    0    0
## [5,]    1    0    0    0    0
## [6,]    0    1    0    0    0
## [7,]    1    0    0    0    0
## [8,]    1    0    0    0    0
## [9,]    0    1    0    0    0
## [10,]   1    0    0    0    0
## [11,]   0    1    0    0    0
## [12,]   0    0    1    0    0
## [13,]   0    0    1    0    0
## [14,]   0    0    0    1    0
## [15,]   0    0    1    0    0
## [16,]   0    0    1    0    0
## [17,]   0    0    1    0    0
## [18,]   0    0    0    1    0
## [19,]   0    0    1    0    0
## [20,]   0    0    1    0    0
## [21,]   0    0    1    0    0
## [22,]   0    0    0    1    0
## [23,]   0    0    1    0    0
## [24,]   0    0    1    0    0
## [25,]   0    0    1    0    0
## [26,]   0    0    0    0    1
```

###matrix5_klamath

#Hypothesis 4: ALL sensors are the same

```
matrix5_klamath <- matrix(nrow=26, ncol=1)
matrix5_klamath[,] <- 1
matrix5_klamath
```

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

```
## [2,] 1
## [3,] 1
## [4,] 1
## [5,] 1
## [6,] 1
## [7,] 1
## [8,] 1
## [9,] 1
## [10,] 1
## [11,] 1
## [12,] 1
## [13,] 1
## [14,] 1
## [15,] 1
## [16,] 1
## [17,] 1
## [18,] 1
## [19,] 1
## [20,] 1
## [21,] 1
## [22,] 1
## [23,] 1
## [24,] 1
## [25,] 1
## [26,] 1
```

```
saveRDS(matrix2_klamath,"matrix2_klamath.rds")
saveRDS(matrix3_klamath,"matrix3_klamath.rds")
saveRDS(matrix4_klamath,"matrix4_klamath.rds")
saveRDS(matrix5_klamath,"matrix5_klamath.rds")
```

##MARSS models ###model 1

#Hypothesis 1, Model 1: all separate

```
mod11_klamath = list()
mod11_klamath$A = "zero" #no trend because we z scored
mod11_klamath$Z = matrix2_klamath
mod11_klamath$R = "diagonal and equal" #all the sensors are same, so observation error should be same
mod11_klamath$Q = "diagonal and unequal"
mod11_klamath$B = "identity" #assuming no species interactions
mod11_klamath$U = "zero" #no trend because we z scored
mod11_klamath$C = "unequal" #Can set C to unequal because it is going off the Z matrix where I have already indicated how to split up the sites.
mod11_klamath$c = matrixc_klamath
mod11_klamath.fit = MARSS(transformed_dat_klamath, model=mod11_klamath, control=list(maxit=10000))
```

```
## Success! abstol and log-log tests passed at 65 iterations.
## Alert: conv.test.slope.tol is 0.5.
## Test with smaller values (<0.1) to ensure convergence.
```

```

##
## MARSS fit is
## Estimation method: kem
## Convergence test: conv.test.slope.tol = 0.5, abstol = 0.001
## Estimation converged in 65 iterations.
## Log-likelihood: 1566.017
## AIC: -3058.035   AICc: -3057.714
##
##               Estimate
## R.diag        3.09e-02
## Q.(X1,X1)      5.35e-03
## Q.(X2,X2)      7.28e-03
## Q.(X3,X3)      7.68e-03
## Q.(X4,X4)      6.63e-03
## Q.(X5,X5)      5.44e-03
## Q.(X6,X6)      1.62e-02
## Q.(X7,X7)      4.93e-03
## Q.(X8,X8)      1.77e-03
## Q.(X9,X9)      2.59e-03
## Q.(X10,X10)    2.21e-03
## Q.(X11,X11)    1.00e-02
## Q.(X12,X12)    5.10e-03
## x0.X1          1.39e+00
## x0.X2          9.89e-01
## x0.X3          1.10e+00
## x0.X4          1.22e+00
## x0.X5          1.54e+00
## x0.X6          1.42e+00
## x0.X7          8.31e-01
## x0.X8          6.65e-01
## x0.X9          1.03e+00
## x0.X10         1.01e+00
## x0.X11         1.05e+00
## x0.X12         9.96e-01
## C.X1           2.05e-04
## C.X2           4.26e-04
## C.X3           4.58e-04
## C.X4           2.04e-04
## C.X5           -9.02e-05
## C.X6           9.00e-05
## C.X7           2.43e-04
## C.X8           5.48e-04
## C.X9           3.73e-04
## C.X10          3.66e-04
## C.X11          6.39e-04
## C.X12          2.62e-04
## Initial states (x0) defined at t=0
##
## Standard errors have not been calculated.
## Use MARSSparamCIs to compute CIs and bias estimates.

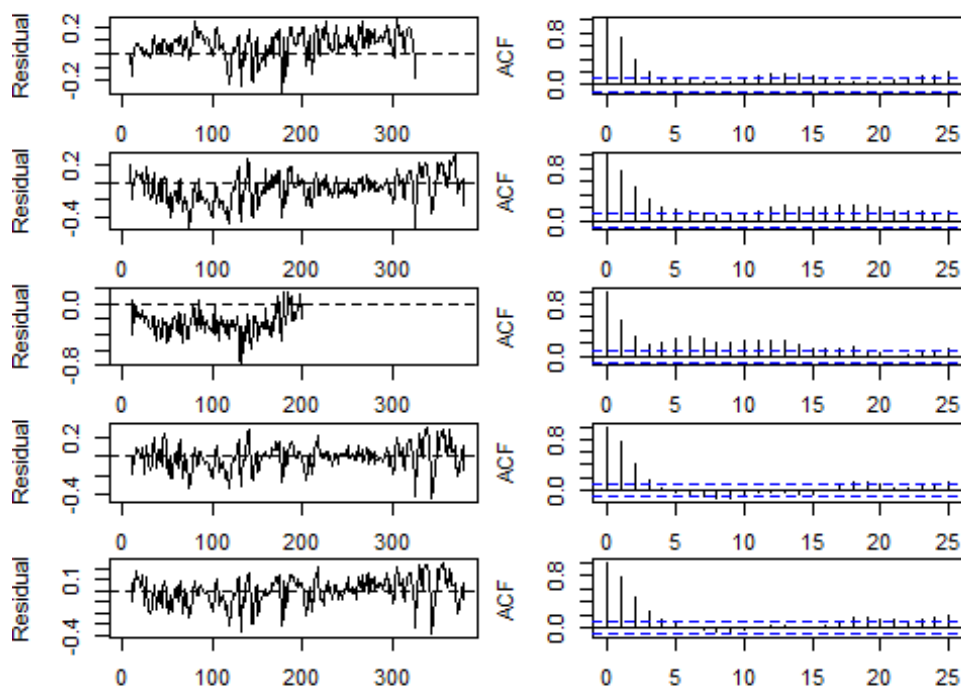
```

```

mod11_klamath.params = MARSSparamCIs(mod11_klamath.fit)
saveRDS(mod11_klamath.fit, "mod11_klamath.fit.rds")
saveRDS(mod11_klamath.params, "mod11_klamath.params.rds")

par(mfrow=c(5,2), mai=c(0.1,0.5,0.2,0.1), omi=c(0.5,0,0,0))
for (j in 1:5) {
  plot.ts(residuals<-MARSSresiduals(mod11_klamath.fit, type = "tt1")$model.
residuals[j, ],
        ylab = "Residual")
  abline(h = 0, lty = "dashed")
  acf(residuals, na.action = na.pass)
}

```



###model 2

#Hypothesis 2, Model 3: ponds vs. creeks

```

mod12_klamath <- mod11_klamath
mod12_klamath$Z <- matrix3_klamath
mod12_klamath.fit = MARSS(transformed_dat_klamath, model=mod12_klamath, contr
ol=list(maxit=10000))

```

```

## Success! abstol and log-log tests passed at 48 iterations.
## Alert: conv.test.slope.tol is 0.5.
## Test with smaller values (<0.1) to ensure convergence.
##
## MARSS fit is
## Estimation method: kem

```



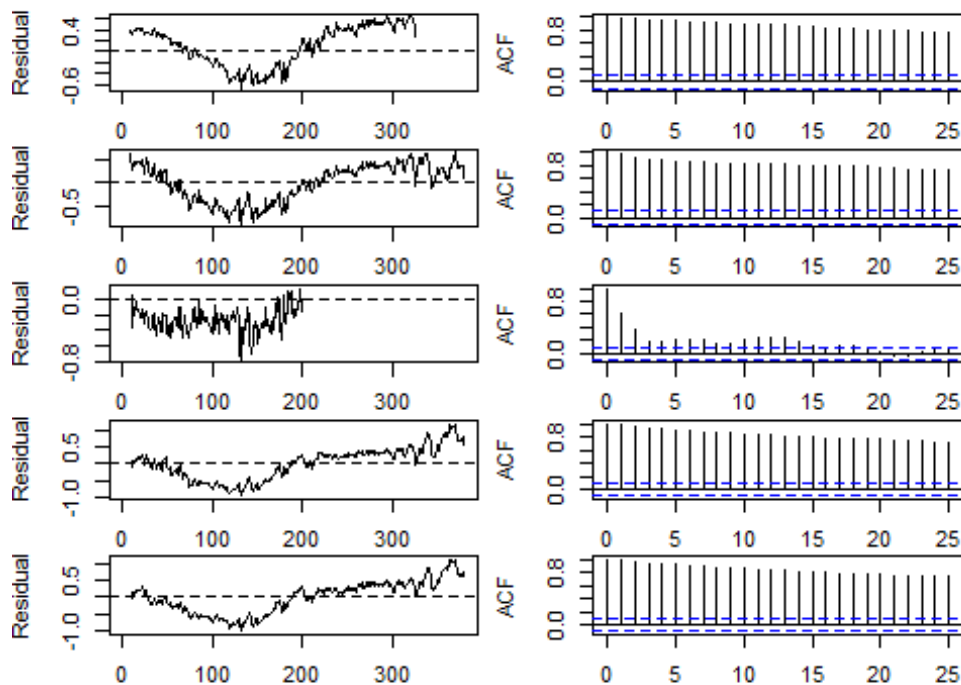
```

## Convergence test: conv.test.slope.tol = 0.5, abstol = 0.001
## Estimation converged in 48 iterations.
## Log-likelihood: -4126.332
## AIC: 8272.665    AICc: 8272.69
##
##           Estimate
## R.diag    0.141571
## Q.(X1,X1) 0.002861
## Q.(X2,X2) 0.006578
## Q.(X3,X3) 0.004944
## x0.X1     0.972943
## x0.X2     1.090202
## x0.X3     1.034314
## C.X1      0.000368
## C.X2      0.000429
## C.X3      0.000158
## Initial states (x0) defined at t=0
##
## Standard errors have not been calculated.
## Use MARSSparamCIs to compute CIs and bias estimates.

mod12_klamath.params = MARSSparamCIs(mod12_klamath.fit)
saveRDS(mod12_klamath.fit, "mod12_klamath.fit.rds")
saveRDS(mod12_klamath.params, "mod12_klamath.params.rds")

par(mfrow=c(5,2), mai=c(0.1,0.5,0.2,0.1), omi=c(0.5,0,0,0))
for (j in 1:5) {
  plot.ts(residuals<-MARSSresiduals(mod12_klamath.fit, type = "tt1")$model.
residuals[j, ],
        ylab = "Residual")
  abline(h = 0, lty = "dashed")
  acf(residuals, na.action = na.pass)
}

```



###model 3

#Hypothesis 3, Model 3: Tributary versus tributary

```
mod13_klamath <- mod11_klamath
mod13_klamath$Z = matrix4_klamath
mod13_klamath.fit = MARSS(transformed_dat_klamath, model=mod13_klamath, control=list(maxit=10000))
```

Success! abstol and log-log tests passed at 53 iterations.

Alert: conv.test.slope.tol is 0.5.

Test with smaller values (<0.1) to ensure convergence.

##

MARSS fit is

Estimation method: kem

Convergence test: conv.test.slope.tol = 0.5, abstol = 0.001

Estimation converged in 53 iterations.

Log-likelihood: -1045.504

AIC: 2123.008 AICc: 2123.069

##

Estimate

R.diag 0.066270

Q.(X1,X1) 0.006797

Q.(X2,X2) 0.005902

Q.(X3,X3) 0.001860

Q.(X4,X4) 0.008371

Q.(X5,X5) 0.004792

x0.X1 1.259193

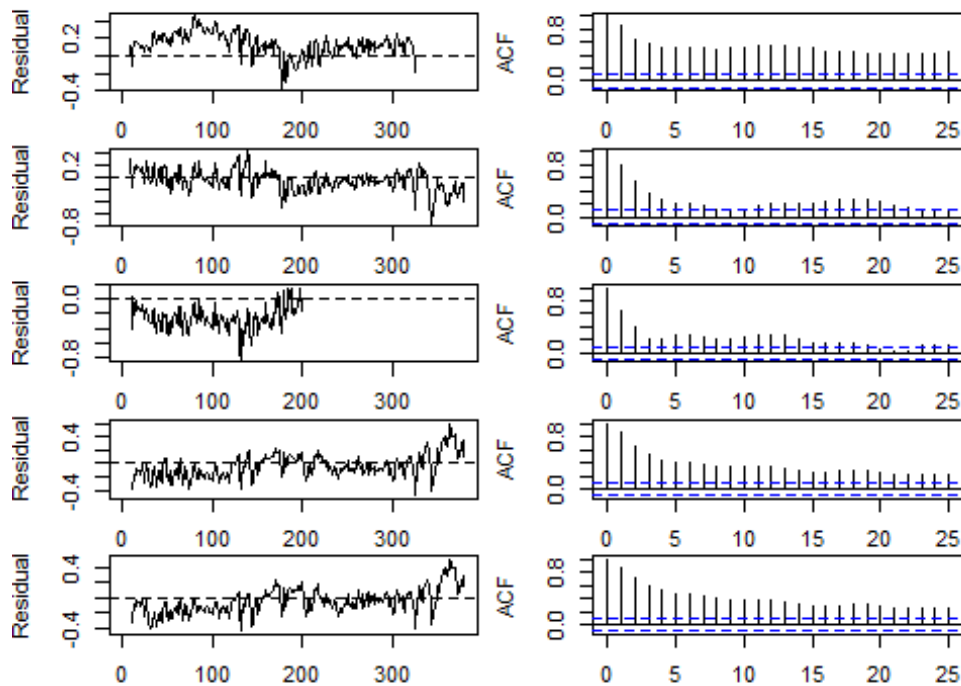
```

## x0.X2      1.138327
## x0.X3      0.939603
## x0.X4      1.067854
## x0.X5      1.007397
## C.X1       0.000275
## C.X2       0.000357
## C.X3       0.000390
## C.X4       0.000519
## C.X5       0.000202
## Initial states (x0) defined at t=0
##
## Standard errors have not been calculated.
## Use MARSSparamCIs to compute CIs and bias estimates.

mod13_klamath.params = MARSSparamCIs(mod13_klamath.fit)
saveRDS(mod13_klamath.fit,"mod13_klamath.fit.rds")
saveRDS(mod13_klamath.params,"mod13_klamath.params.rds")

par(mfrow=c(5,2), mai=c(0.1,0.5,0.2,0.1), omi=c(0.5,0,0,0))
for (j in 1:5) {
  plot.ts(residuals<-MARSSresiduals(mod13_klamath.fit, type = "tt1")$model.
residuals[j, ],
          ylab = "Residual")
  abline(h = 0, lty = "dashed")
  acf(residuals,na.action = na.pass)
}

```



###model 4

#Hypothesis 4, Model 4: All same

```
mod14_klamath <- mod11_klamath
```

```
mod14_klamath$Z <- matrix5_klamath
```

```
mod14_klamath.fit = MARSS(transformed_dat_klamath, model=mod14_klamath, control=list(maxit=10000))
```

```
## Success! abstol and log-log tests passed at 21 iterations.
```

```
## Alert: conv.test.slope.tol is 0.5.
```

```
## Test with smaller values (<0.1) to ensure convergence.
```

```
##
```

```
## MARSS fit is
```

```
## Estimation method: kem
```

```
## Convergence test: conv.test.slope.tol = 0.5, abstol = 0.001
```

```
## Estimation converged in 21 iterations.
```

```
## Log-likelihood: -4232.919
```

```
## AIC: 8473.839   AICc: 8473.843
```

```
##
```

```
##      Estimate
```

```
## R.diag 0.148267
```

```
## Q.Q    0.003658
```

```
## x0.x0  0.975406
```

```
## C.C    0.000435
```

```
## Initial states (x0) defined at t=0
```

```
##
```

```
## Standard errors have not been calculated.
```

```
## Use MARSSparamCIs to compute CIs and bias estimates.
```

```
mod14_klamath.params =MARSSparamCIs(mod14_klamath.fit)
```

```
saveRDS(mod14_klamath.fit,"mod14_klamath.fit.rds")
```

```
saveRDS(mod14_klamath.params,"mod14_klamath.params.rds")
```

```
par(mfrow=c(5,2), mai=c(0.1,0.5,0.2,0.1), omi=c(0.5,0,0,0))
```

```
  for (j in 1:5) {
```

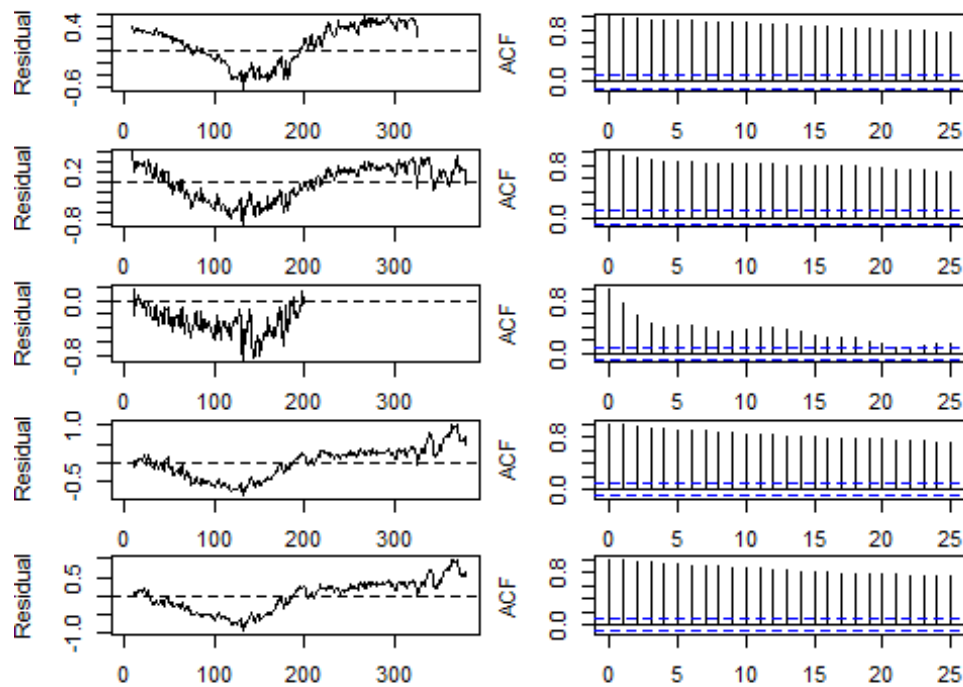
```
    plot.ts(residuals<-MARSSresiduals(mod14_klamath.fit, type = "tt1")$model.  
residuals[j, ],
```

```
          ylab = "Residual")
```

```
    abline(h = 0, lty = "dashed")
```

```
    acf(residuals,na.action = na.pass)
```

```
  }
```



##AICc

```
data.frame(Model=c("Model11_klamath", "Model12_klamath", "Model13_klamath", "
Model14_klamath"),
           AICc=round(c(mod11_klamath.fit$AICc,
                        mod12_klamath.fit$AICc,
                        mod13_klamath.fit$AICc,
                        mod14_klamath.fit$AICc),1))

##           Model    AICc
## 1 Model11_klamath -3057.7
## 2 Model12_klamath  8272.7
## 3 Model13_klamath  2123.1
## 4 Model14_klamath  8473.8
```