Homework 7

Casey Gibson

4/2/2018

Problem 1

$$y_t = \rho y_{t-1} + \epsilon_t$$
$$\gamma_k = Cov(y_{t-k}, y_t) = \rho^k \gamma_0$$
$$y_{t-k} \cdot y_t = y_{t-k} \cdot \rho y_{t-1} + y_{t-k} \cdot \epsilon_t$$
$$E(y_{t-k} \cdot y_t) = \rho E(y_{t-k} \cdot y_{t-1}) + E(y_{t-k}) \cdot E(\epsilon_t)$$

by independence.

$$E(y_{t-k} \cdot y_t) = \rho E(y_{t-k} \cdot y_{t-1})$$

and we can keep going , by writing the right hand side as

$$E(y_{t-k} \cdot y_{t-1}) = \rho E(y_{t-k} \cdot y_{t-2}) + 0$$

Pluggin this into the above we get

$$E(y_{t-k} \cdot y_t) = \rho^2 E(y_{t-k} \cdot y_{t-2})$$

So continuing we see that

$$E(y_{t-k} \cdot y_t) = \rho^k E(y_{t-k} \cdot y_{t-k})$$

We know that

$$E(y_{t-k}^2) = Var(y_{t-k}) - E(y_{t-k})^2$$
$$E(y_{t-k}^2) = Var(y_{t-k}) - 0^2$$

because $E(y_{t-k}) = 0$.

We also know that

$$Var(y_{t-k}) = Cov(y_{t-k}, y_{t-k}) = Cov(y_t, y_t) = \gamma_0$$

So plugging back in we see,

$$E(y_{t-k} \cdot y_t) = \rho^k \gamma_0$$

Problem 2

```r
library(R2jags)
```

```
## Loading required package: rjags
## Loading required package: coda
## Linked to JAGS 4.3.0
## Loaded modules: basemod,bugs
```

```
##
## Attaching package: 'R2jags'
## The following object is masked from 'package:coda':
##
##     traceplot
```

```r
library(rjags)

GetAR <- function(nyears, # length of series
                  rho, sigma, # AR parameters
                  eps0.t = NULL, # innovations (optional)
                  ystart = NULL # starting value y1 (optional)
                  ){
  if (is.null(eps0.t)){
    set.seed(123)
    eps0.t <- rnorm(nyears, 0, 1)
  }
  y.t <- rep(NA, nyears)
  if (is.null(ystart)){
    y.t[1] <- sigma/sqrt(1-rho^2)*eps0.t[1]
  } else {
    y.t[1] <- ystart
  }
  for (t in 2:nyears){
    y.t[t] <- rho*y.t[t-1] + sigma*eps0.t[t]
  }
  return(y.t)
}
rho <- 0.5
sigma <- 1
sigma.y <- 0.5
nyears <- 100
mu.t <- GetAR(nyears, rho, sigma)
set.seed(124)
y.t <- mu.t + rnorm(nyears, 0, sigma.y)


model <- "
model{
mu.t[1] ~ dnorm(0, tau.stat)

tau.stat <- (1-pow(rho,2))/pow(sigma,2)
for (t in 2:(nyears+P)){
 mu.t[t] ~ dnorm(muhat.t[t], tau)
 muhat.t[t] <- rho*mu.t[t-1]
 yhat.t[t] ~ dnorm(muhat.t[t], tau.y)

}
for (t in 1:nyears){
  y.t[t] ~ dnorm(mu.t[t], tau.y)
}


tau <- pow(sigma,-2)
```

```
sigma ~ dunif(0,3)
tau.y <- pow(sigma.y,-2)
sigma.y ~ dunif(0,3)
rho ~ dunif(-1,1)
}
"

P <- 20
set.seed(1234)
t.i <- sort(sample(seq(1,100), size = 70))
y.t[-t.i] <- NA
jags.data <- list(y.t = y.t, nyears=nyears,P=P)
parnames <- c("sigma", "rho","mu.t","sigma.y","muhat.t","yhat.t")

mod <- jags(data = jags.data,
           parameters.to.save= c(parnames),  n.chains = 4, n.burnin = 1000, n.iter = 1000+30000, n.thin
,
           model.file = textConnection(model))
```

```
## module glm loaded
```

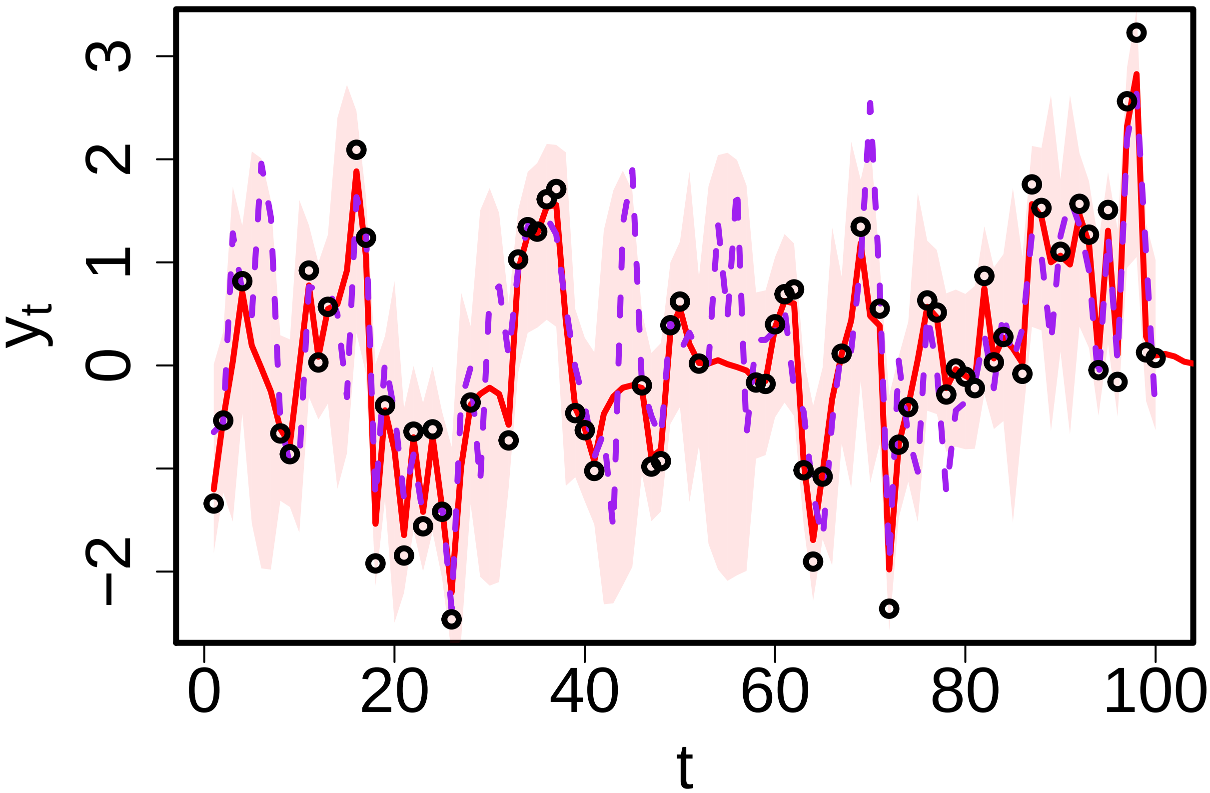
```
## Compiling model graph
##    Resolving undeclared variables
##    Allocating nodes
## Graph information:
##    Observed stochastic nodes: 70
##    Unobserved stochastic nodes: 272
##    Total graph size: 475
##
## Initializing model
```

```
mcmc.array <- mod$BUGSoutput$sims.array
mcmc.list <- mod$BUGSoutput$sims.list
mean(mcmc.array[,,"mu.t[6]"])
```

```
## [1] -0.01087514
```

```
quantile(mcmc.array[,,"mu.t[6]"],c(.025,.975))
```

```
##      2.5%     97.5%
## -1.968996  2.008788
```



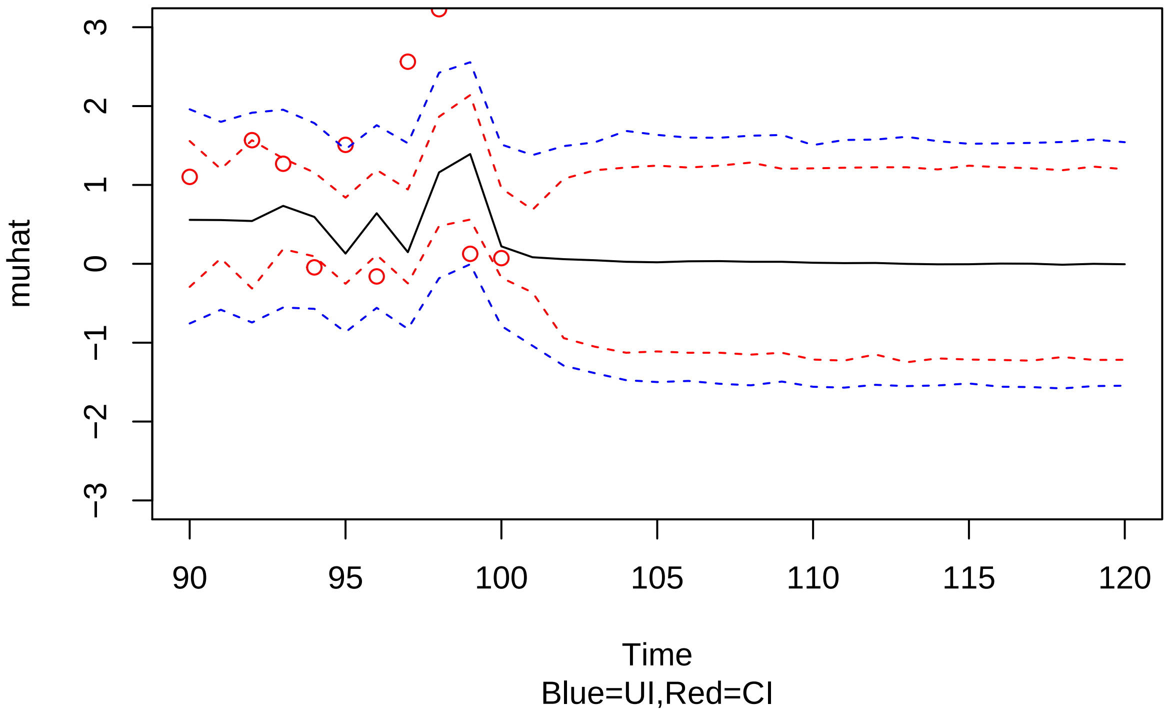
We can forecast μ_t using

$$\mu_{101}^{(s)}|\mu_{100}^{(s)}, \rho^{(s)}, (\delta^2)^{(s)} \sim N(\mu_{100}^{(s)}, (\delta^2)^{(s)})$$

and forecast y_t using

$$y_{101}^{(s)}|\mu_{101}^{(s)}, (\sigma^2)^{(s)} \sim N(\mu_{101}^{(s)}, (\sigma^2)^{(s)})$$

and so on as we extend in time.




```
print ("muhat[101]")
```

```
## [1] "muhat[101]"
```

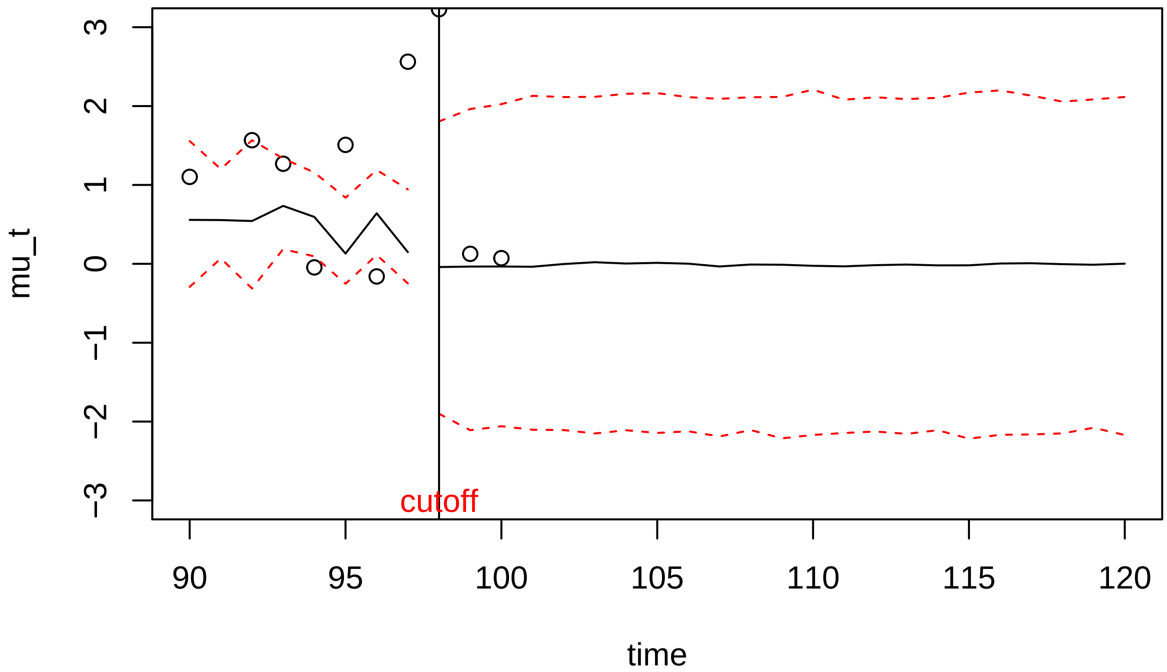
```
print (mean(mcmc.array[,,"muhat.t[101]"]))
```

```
## [1] 0.08308189
```

```
print (quantile(mcmc.array[,,"muhat.t[101]"],c(.025,.975)))
```

```
##       2.5%      97.5%
## -0.3632181  0.6854923
```

c




```r
print ("muhat[98]")
```

```
## [1] "muhat[98]"
```

```r
print (mean(mu.sp[,1]))
```

```
## [1] -0.04101947
```

```r
print (quantile(mu.sp[,1],c(.025,.975)))
```

```
##      2.5%     97.5%
## -1.901172  1.804929
```

The PI's that result from part *b* are smaller than those of part *c* because we are forecasting from later on in the time series.