

KU Leuven Summer School
Segment 2A
First Look at Latents - Missing Data

Paul Gustafson

September 15, 2022

Missing data

```
require(mice) ### just want data ex. from this package
```

```
summary(nhanes2)
```

##	age	bmi	hyp	chl
##	20-39:12	Min. :20.4	no :13	Min. :113
##	40-59: 7	1st Qu.:22.6	yes : 4	1st Qu.:185
##	60-99: 6	Median :26.8	NA's: 8	Median :187
##		Mean :26.6		Mean :191
##		3rd Qu.:28.9		3rd Qu.:212
##		Max. :35.3		Max. :284
##		NA's :9		NA's :10

Say interested in regressing *chl* (Y) on *age* (X_1), *hyp* (X_2), and *bmi* (X_3).

Toward a generative model (1 of 3)

$$f(\theta) \prod_{i=1}^n f(x_{1i}|\theta) f(x_{2i}|x_{1i}, \theta) f(x_{3i}|x_{1i}, x_{2i}, \theta) f(y_i|x_{1i}, x_{2i}, x_{3i}, \theta)$$

Toward a generative model (2 of 3)

```
statmod.string <-"
  for (i in 1:n) {
    x2[i] ~ dbern(pr.x2[i])
    logit(pr.x2[i]) <- alpha0 + alpha1a*x1a[i] +
                      alpha1b*x1b[i]

    x3[i] ~ dnorm(mn.x3[i], prec.x3)
    mn.x3[i] <- kappa0 + kappa1a*x1a[i] +
                kappa1b*x1b[i]+kappa2*x2[i]

    y[i] ~ dnorm(mn.y[i], prec.y)
    mn.y[i] <- beta0 + beta1a*x1a[i] + beta1b*x1b[i] +
                beta2*x2[i] + beta3*x3[i]
  }
"
```

Toward generative model (3 of 3)

```
prior.string <- "  
  alpha0 ~ dnorm(0, 0.1)  
  alpha1a ~ dnorm(0, 0.1)  
  alpha1b ~ dnorm(0, 0.1)  
  
  kappa0 ~ dnorm(0, 0.01)  
  kappa1a ~ dnorm(0, 0.01)  
  kappa1b ~ dnorm(0, 0.01)  
  kappa2 ~ dnorm(0, 0.01)  
  prec.x3 ~ dgamma(0.1, 0.1)  
  
  beta0 ~ dnorm(0, 0.01)  
  beta1a ~ dnorm(0, 0.01)  
  beta1b ~ dnorm(0, 0.01)  
  beta2 ~ dnorm(0, 0.01)  
  beta3 ~ dnorm(0, 0.01)  
  prec.y ~ dgamma(0.5, 0.5)  
  sig.y <- sqrt(1/prec.y)  
"
```

Housekeeping

```
genmod.string <- paste(  
  "model {",prior.string, statmod.string,"}")
```

Pause to comment on this prior and stat model specification

Almost have supplied a joint distribution of everything

Turn the JAGS crank

```
### generative model, data go in
mod <- jags.model(textConnection(genmod.string),
  data=list(x1a=as.numeric(nhanes2$age=="40-59"),
    x1b=as.numeric(nhanes2$age=="60-99"),
    x2=as.numeric(nhanes2$hyp)-1,
    x3=nhanes2$bmi,
    y=nhanes2$chl,
    n=dim(nhanes2)[1]),
  n.chains=4)

update(mod, 2000)    ### burn-in

### MC output comes out
opt1.JAGS <- coda.samples(mod, n.iter=10000,
  variable.names=c("beta1a", "beta1b", "beta2", "beta3", "sig.y",
    "x2[6]", "x3[6]", "y[6]"))
```


JAGS, continued

```
summary(opt1.JAGS)
```

```
##
## Iterations = 3001:13000
## Thinning interval = 1
## Number of chains = 4
## Sample size per chain = 10000
##
## 1. Empirical mean and standard deviation for each variable,
##    plus standard error of the mean:
##
##           Mean      SD Naive SE Time-series SE
## beta1a    5.427 9.354  0.04677      0.05238
## beta1b    7.789 9.838  0.04919      0.05583
## beta2     4.614 9.455  0.04728      0.05084
## beta3     6.886 0.586  0.00293      0.00486
## sig.y    43.158 9.252  0.04626      0.05684
## x2[6]      0.431 0.495  0.00248      0.00339
## x3[6]     24.807 3.992  0.01996      0.02557
## y[6]     184.000 0.000  0.00000      0.00000
##
## 2. Quantiles for each variable:
```

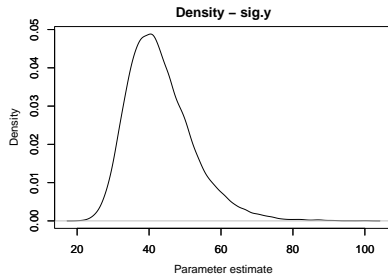
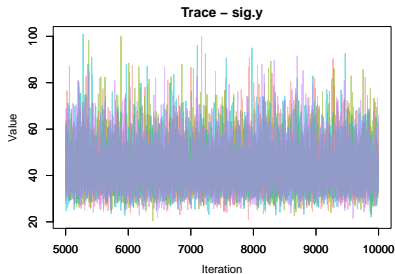
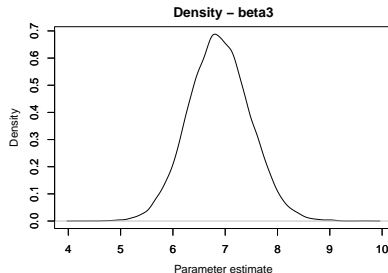
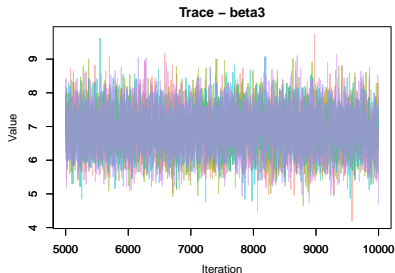
Or

```
require(MCMCvis)
MCMCsummary(opt1.JAGS)
```

##	mean	sd	2.5%	50%	97.5%	Rhat	n.eff
## beta1a	5.427	9.354	-13.00	5.43	23.67	1	31926
## beta1b	7.789	9.838	-11.54	7.87	27.06	1	31076
## beta2	4.614	9.455	-13.96	4.65	23.04	1	34592
## beta3	6.886	0.586	5.76	6.88	8.05	1	14580
## sig.y	43.158	9.252	29.12	41.77	65.05	1	26513
## x2[6]	0.431	0.495	0.00	0.00	1.00	1	21329
## x3[6]	24.807	3.992	17.01	24.78	32.77	1	24376
## y[6]	184.000	0.000	184.00	184.00	184.00	NaN	0

Some due diligence on our computational work

```
MCMCtrace(opt1.JAGS, params=c("beta3","sig.y"), pdf=F)
```



Thoughts: Under-the-hood, exactly what distribution are the Monte Carlo draws (approximately) coming from?

Thoughts: Hall-pass that freed me from having a model for X_1 ?

More space for thoughts?

Thoughts: In concept at least could I have done the computing in the “collapsed” frame of reference?

$$f(\theta) \prod_{i=1}^n f(\text{observed}_i | \theta)$$

Collapsed, for instance:

```
nhanes2[c(3,6),]
```

```
##      age bmi  hyp chl  
## 3 20-39  NA   no 187  
## 6 60-99  NA <NA> 184
```


Collapsed versus Augmented: Two different strategies **for computing the same thing**

Collapsed:

$$f(\theta|\text{observed}) \propto f(\text{observed}|\theta)f(\theta)$$

Augmented:

$$f(\theta|\text{observed}) = \int f(\theta, \text{latent}|\text{observed})d\text{latent}$$

Any more thoughts?