

# Bayesian method



# Definition

- Bayesian statistics, named for Thomas Bayes (1701–1761), is a theory in the field of statistics in which the evidence about the true state of the world is expressed in terms of degrees of belief known as Bayesian probabilities.
- The formulation of statistical models using Bayesian statistics has the unique feature of requiring the specification of prior distributions for any unknown parameters. These prior distributions are as integral to a Bayesian approach to statistical modelling as the expression of probability distributions.
- The Bayesian design of experiments includes a concept called 'influence of prior beliefs'. This approach uses sequential analysis techniques to include the outcome of earlier experiments in the design of the next experiment. This is achieved by updating 'beliefs' through the use of prior and posterior distribution.

# Fundamental difference Bayesian and frequentist approaches

## Bayesian Approach

- Inference of  $\theta \rightarrow$  based on  $p(\theta | y)$
- Inference of  $\ddot{Y} \rightarrow$  based on  $p(\ddot{Y} | y)$

## Frequentist Approach

- Inference of  $\theta \rightarrow$  based on  $p(y | \theta)$
- Inference of  $\ddot{Y} \rightarrow$  based on  $\theta \rightarrow$  based on  $p(y | \theta)$

## Compare Bayesian vs. frequentist approaches

- Bayesian statistical conclusions: using probability statements ('highly unlikely', 'very likely'); frequentist statistical conclusions: using p-values ('significant', 'test can not be rejected', etc.)
- Results obtained using the two different procedures yield superficially similar results (especially in asymptotic cases)
- Bayesian methods can be easily extended to more complex problems
- Usually Bayesian models work better with less data

# Bayesian method

(Thomas Bayes 1763)

Bayesian: Probability (parameter, given data)

Frequentist: Probability (data, given parameter)

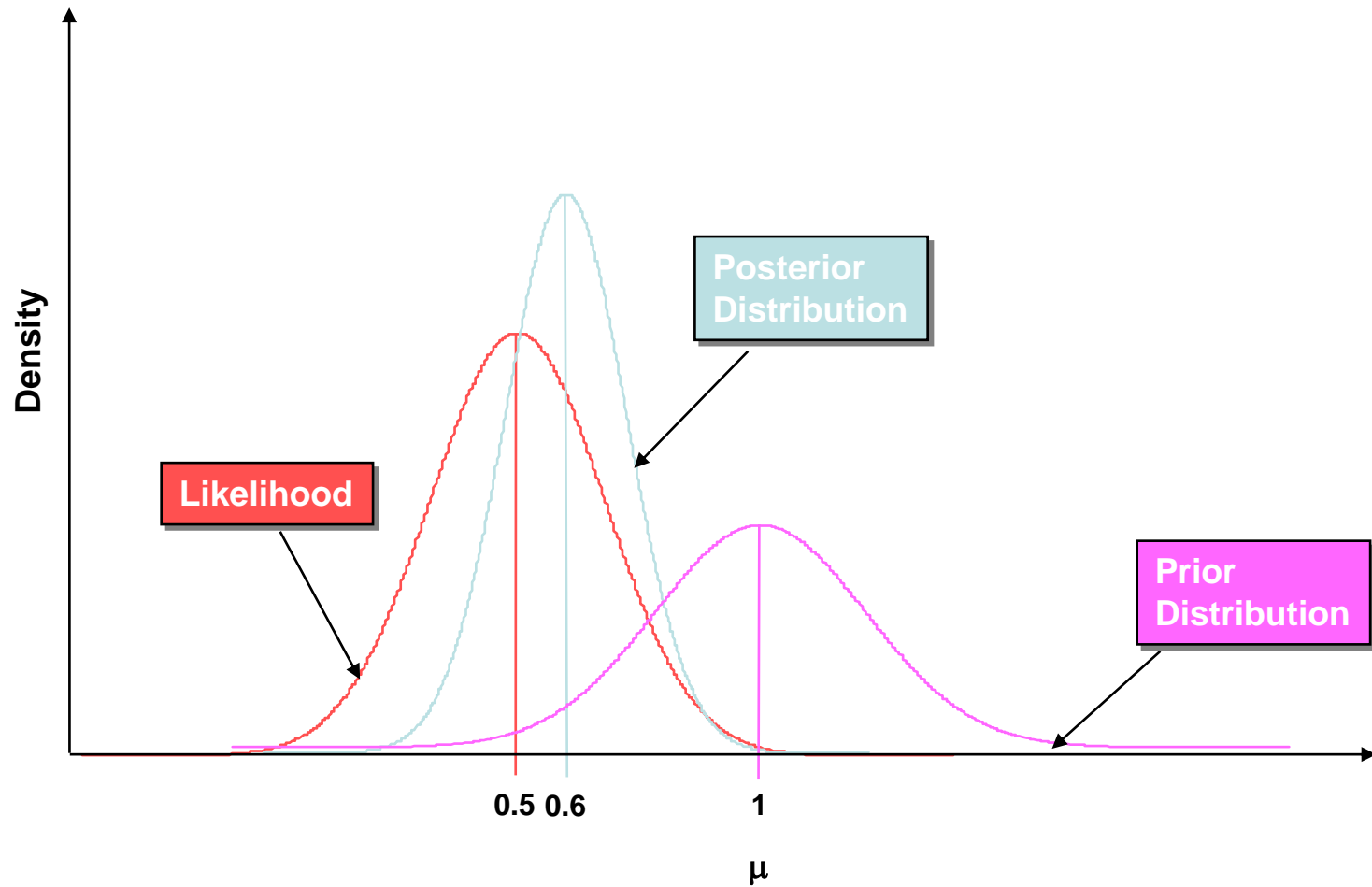
## Bayes' Theorem

The diagram illustrates Bayes' Theorem with the following components and labels:

- Posterior probability:** Labeled with an arrow pointing to  $P(\theta|y)$ .
- Likelihood:** Labeled with an arrow pointing to  $f(y|\theta)$ .
- Prior probability:** Labeled with an arrow pointing to  $P(\theta)$ .
- Marginal probability of y:** Labeled with an arrow pointing to  $m(y)$ .

$$P(\theta|y) = \frac{f(y|\theta) P(\theta)}{m(y)}$$
$$m(y) = \int f(y|\theta) p(\theta) d\theta$$

# Prior and posterior distributions



# The prior distribution

- Prior distribution of  $\theta$ : “subjective” probability that quantifies whatever belief (however vague), we may have about a certain value before having looked at the data.

## **Bayesian method is appropriate for dynamics, nonlinear, and full-of-noise ecological processes**

Systems invariably driven by endogenous dynamic processes plus demographic and environmental process noise, and are only observable with error.

The inability to make well-founded statistical inferences about biological dynamic models in the chaotic and near-chaotic regimes, ..., leaves dynamic theory without the methods of quantitative validation that are essential tools in the rest of biological science.

Here I show that this impasse can be resolved in a simple and general manner, ..., using a straightforward Markov chain Monte Carlo (MCMC) sampler (Wood 2010).

Wood SN (2010). Statistical inference for noisy nonlinear ecological dynamic systems.  
**Nature** 466: 1102-1113.



# Example: Hemophilia Inheritance

- Father (XY), Mother (XX)
- Hemophilia exhibits X-chromosome-linked recessive inheritance
- If son receives a bad chromosome from mother, he will be affected
- If daughter receives one bad chromosome from mother, she will not be affected, but will be a carrier
- If both X are affected in a woman it is fatal (occurrence rare)

## The probability of a mother being a carrier

A woman has an affected son  $\rightarrow$  mother is a carrier of hemophilia

Mother ( $X_{\text{neg}}$   $X_{\text{pos}}$ )

Father (XY) usually not affected

### Unknown quantity of interest

$\theta = 0$  if woman is not a carrier

1 if woman is carrier

**Prior:**  $P(\theta=0) = P(\theta=1) = 0.5$

## Model and likelihood

The woman has two unaffected sons.

Let  $y_i = 1$  denote an affected son

0 denote an unaffected son

**The two conditions of two sons are independent given  $\theta$   
(no two are identical twins).**

$$\Pr(y_1=0, y_2=0 \mid \theta=1) = (0.5)(0.5) = 0.25$$

$$\Pr(y_1=0, y_2=0 \mid \theta=0) = (1)(1) = 1$$

# Posterior distribution

$$P(\theta|y) = \frac{f(y|\theta) P(\theta)}{m(y)}$$

**Bayes Rule:** Combines the information in the data with the prior probability

$y = (y_1, y_2)$  joint data

Posterior probability of a mother being a carrier:

$$p(\theta=1|y)$$

$$= \mathbf{p(y | \theta=1)p(\theta=1) / \{p(y|\theta=1)p(\theta=1) + p(y|\theta=0)p(\theta=0)\}}$$

$$= (0.25)(0.5) / \{(0.25)(0.5) + (1)(0.5)\}$$

$$= \frac{0.25 \times 0.5}{0.25 \times 0.5 + 1 \times 0.5} = 0.2$$

# Conclusions

- It is clear that if the woman has unaffected children it is less probable she is a carrier
- Bayes rule provides a formal mechanism in terms of prior and posterior odds.

# Easy sequential analysis performance with Bayesian analysis

Suppose that the woman has a third son, also unaffected.

Use previous posterior distribution as new prior

$$\begin{aligned} & \mathbf{P}(\theta=1 \mid \mathbf{y}_1, \mathbf{y}_2, \mathbf{y}_3) \\ &= P(y_3 \mid \theta=1)(0.2) / \{ P(y_3 \mid \theta=1)(0.2) + P(y_3 \mid \theta=0)(0.8) \} \\ &= (0.5)(0.2) / \{ (0.5)(0.2) + (1)(0.8) \} \\ &= 0.111 \end{aligned}$$

## Probability of having a rare disease

$$P(\theta|y) = \frac{f(y|\theta) P(\theta)}{m(y)}$$

Tom had received a health check report indicating the result of a rare disease is positive. The probability of this disease in regular population is 0.1%. The checking machine usually gives 1% false positive reports.

We define:  $\theta = 1$  if Tom has the disease, and  $\theta = 0$  if Tom has not.  
 $y$  is the probability of having a positive result.

Posterior probability of Tom having the disease,  $p(\theta=1|y)$

$$= \mathbf{p(y|\theta=1)p(\theta=1) / \{p(y|\theta=1)p(\theta=1) + p(y|\theta=0)p(\theta=0)\}}$$

$$= \frac{0.99 \times 0.001}{0.99 \times 0.001 + 0.01 \times 0.999} = 0.090$$

If the second check still gives a positive result, the probability  $p(\theta=1|y)$  is:

$$= \frac{0.99 \times 0.09}{0.99 \times 0.09 + 0.01 \times 0.91} = 0.907$$

## Posterior simulation: Markov Chain Monte Carlo (MCMC)

- **Definition:** A *Markov chain* is a sequence of random variables  $\theta^1, \theta^2, \dots$ , for which, for any  $t$ , the distribution of  $\theta^t$  given all previous  $\theta$ 's depends only on the most recent value,  $\theta^{t-1}$ .
- **Key:** Create a Markov process whose stationary distribution is the specified  $p(\theta|y)$ , and run the simulation long enough that the distribution of the current draws is close enough to this stationary distribution.
- For any specific  $p(\theta|y)$ , a **variety** of Markov chains with the desired property can be constructed



# Algorithms for MCMC

Two popular algorithms:

1. Gibbs sampler
2. Metropolis-Hastings algorithm

Obtaining a sequence of random samples from a probability distribution for which direct sampling is difficult. This sequence can be used to approximate the distribution (e.g., to generate a histogram), or to compute an integral (such as an expected value).

# The Metropolis–Hastings algorithm

Named after Nicholas Metropolis and W. K. Hastings. The Metropolis–Hastings algorithm is an adaptation of a random walk that uses an acceptance/rejection rule to converge to the specified target distribution.

1. Draw a **starting point**  $\theta_0$ , for which  $p(\theta_0|y) > 0$ , from a starting distribution  $p_0(\theta)$ .
2. For  $t=1, 2, \dots$ 
  - (a) Sample a proposal  $\theta^*$  from a jumping distribution at time  $t$ ,  $J_t(\theta^*|\theta_{t-1})$ . For the Metropolis algorithm,  $J_t(\theta_a|\theta_b) = J_t(\theta_b|\theta_a)$  for all  $\theta_a, \theta_b$ .
  - (b) Calculate the ratio of the densities:  $r = \frac{p(\theta^*|y)}{p(\theta^{t-1}|y)}$
  - (c) If  $r \geq 1$ , then  $\theta^*$  is more likely than  $\theta^{t-1}$ ; automatically accept  $\theta^*$ . Otherwise, accept  $\theta^*$  with the odds  $r$ ; if  $\theta^*$  is rejected, use  $\theta^{t-1}$ .

# The Gibbs Sampler

Named after the physicist Josiah Willard Gibbs, a special case of the Metropolis–Hastings algorithm.

Suppose the parameter vector  $\theta$  has been divided into  $d$  components or subvectors,

$$\theta = (\theta_1, \dots, \theta_d).$$

**In iteration  $t$ , we simulate  $\theta_j^t \sim p(\theta_j | \theta_{-j}^{t-1}, y)$**

where  $\theta_{-j}^{t-1}$  represents all the components of  $\theta$ , except for  $\theta_j$ , at their current values

for  $j=1, \dots, d$

# Sample linear regression

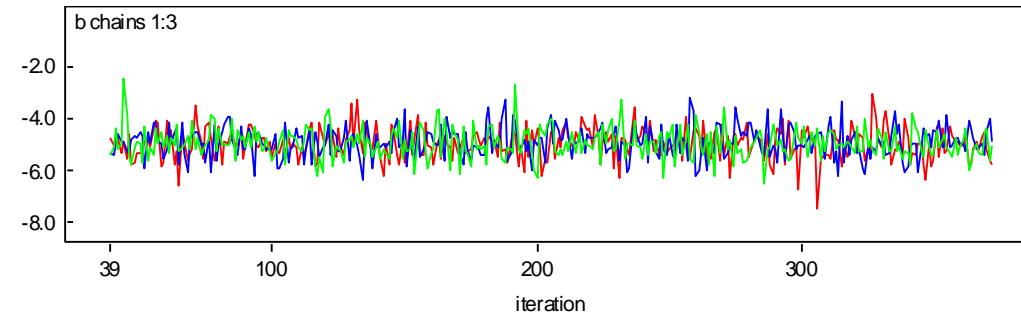
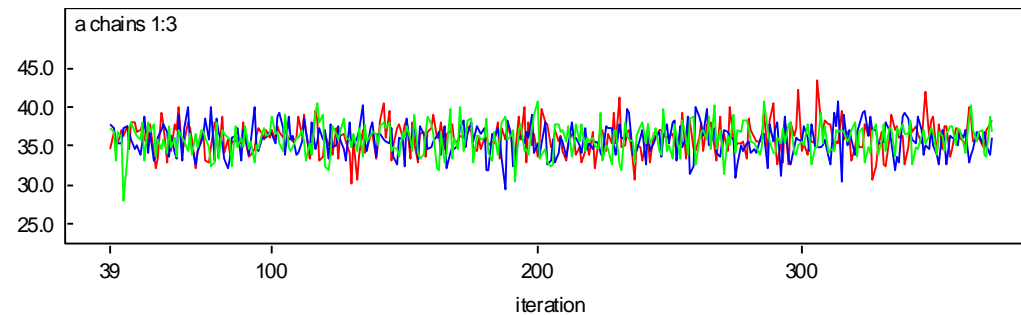
```
X = mtcars$wt; Y = mtcars$mpg; N = nrow(mtcars)
```

```
library(R2WinBUGS)
# WinBUGS code
sink("d:/text/ibis_Bayes/reg.txt")
cat("
  model {
    for(i in 1:N){
      Y[i] ~ dnorm(mu[i], tau)
      mu[i] <- a + b*X[i]
    }
    a ~ dnorm(0,.01)
    b ~ dnorm(0,.01)
    tau ~ dgamma(0.01, 0.01)
  }
", fill = TRUE)
sink()
```

```
# R code
data <- list ( "Y", "X", "N")
inits <- function() list(a=1, b=1, tau=1)
parameters <- c("a", "b", "tau")
```

```
out <- bugs(data, inits, parameters, "d:/text/ibis_Bayes/reg.txt",
  n.chain=3, n.burnin=1000, n.iter=10000, debug=T,
  bugs.directory = "d:/softwares/WinBUGS14/")
```

node	mean	sd	MC error	2.5%	median	97.5%	start	sample
a	35.88	1.94	0.05797	32.25	35.94	39.71	39	1002
b	-4.947	0.5799	0.01685	-6.063	-4.978	-3.832	39	1002
deviance	163.8	3.054	0.09096	160.4	163.1	171.7	39	1002
tau	0.1062	0.02838	9.475E-4	0.05891	0.1036	0.169	39	1002



## A simple example: mortality of moths exposed to cypermethrin

(Royle and Dorazio 2008 Page 66)

The experiment was designed to test whether males and females moths suffered the same mortality when exposed to identical doses of cypermethrin (氯氰菊酯).

Data observed in a dose-response experiment involving adults of the tobacco budworm (*Heliothis virescens*, 烟青虫), a moth species whose larvae are responsible for damage to cotton crops in the United States and Central and South America (Collett, 1991, Example 3.7).

In the experiment, batches of 20 moths of each sex were exposed to a pesticide called cypermethrin for a period of 72 hours, beginning two days after the adults had emerged from pupation. Both sexes were exposed to the same range of pesticide doses: 1, 2, 4, 8, 16, and 32 g cypermethrin.

At the end of the experiment the number of dead moths in each batch were recorded.

y	sex	sexcode	dose
1	male	1	1
4	male	1	2
9	male	1	4
13	male	1	8
18	male	1	16
20	male	1	32
0	female	0	1
2	female	0	2
6	female	0	4
10	female	0	8
12	female	0	16
16	female	0	32

# Models

Let  $x_i$  denote the  $\log_2(\text{dose})$  of cypermethrin administered to the  $i$ th batch of moths that contained either males ( $z_i = 1$ ) or females ( $z_i = 0$ ). Each batch has  $N = 20$  moths. A logistic-regression model containing 3 parameters is:

$$y_i | N, p_i \sim \text{Bin}(N, p_i)$$

$$\text{logit}(p_i) = \alpha + \beta x_i + \gamma z_i$$

y	sex	sexcode	dose
1	male	1	1
4	male	1	2
9	male	1	4
13	male	1	8
18	male	1	16
20	male	1	32
0	female	0	1
2	female	0	2
6	female	0	4
10	female	0	8
12	female	0	16
16	female	0	32

where  $\alpha$  is the intercept,  $\beta$  is the effect of cypermethrin and  $\gamma$  is the effect of sex.

# Codes

```
# -----data-----
N = 20 # 每组烟青虫的数量，雌雄各6组
y = c(1,4,9,13,18,20, 0,2,6,10,12,16) # 每组死亡的烟青虫数量
sex = c(rep('male',6), rep('female',6))
dose = rep(c(1,2,4,8,16,32), 2) # 农药剂量
ldose = log(dose)/log(2) # 对数转换
sexcode = rep(0,length(sex)) # 定义性别代码为0
i = sex=='male' # 区分雌雄 (FALSE-TRUE)
sexcode[i] = 1 # 定义雄性代码为1，剩下的雌性依旧为0
as.data.frame(cbind(y, sex, sexcode, dose)) # 显示数据
```

```
# -----arguments for R2WinBUGS-----
data = list(n=length(y), N=N, y=y, x=ldose, z=sexcode)
params = list('alpha', 'beta', 'gamma')
inits = function() {
  list(alpha=rnorm(1), beta=rnorm(1),
        gamma=rnorm(1))
}
```

```
# -----native WinBUGS code-----
modelName = 'd:/code/bugs/model.txt'
cat('
  model {
    alpha ~ dnorm(0, 0.01)
    beta ~ dnorm(0, 0.01)
    gamma ~ dnorm(0, 0.01)
    for (i in 1:n) {
      y[i] ~ dbin(p[i], N)
      logit(p[i]) <- alpha + beta*x[i] + gamma*z[i]
    }
  }
', fill = TRUE, file = modelName)
```

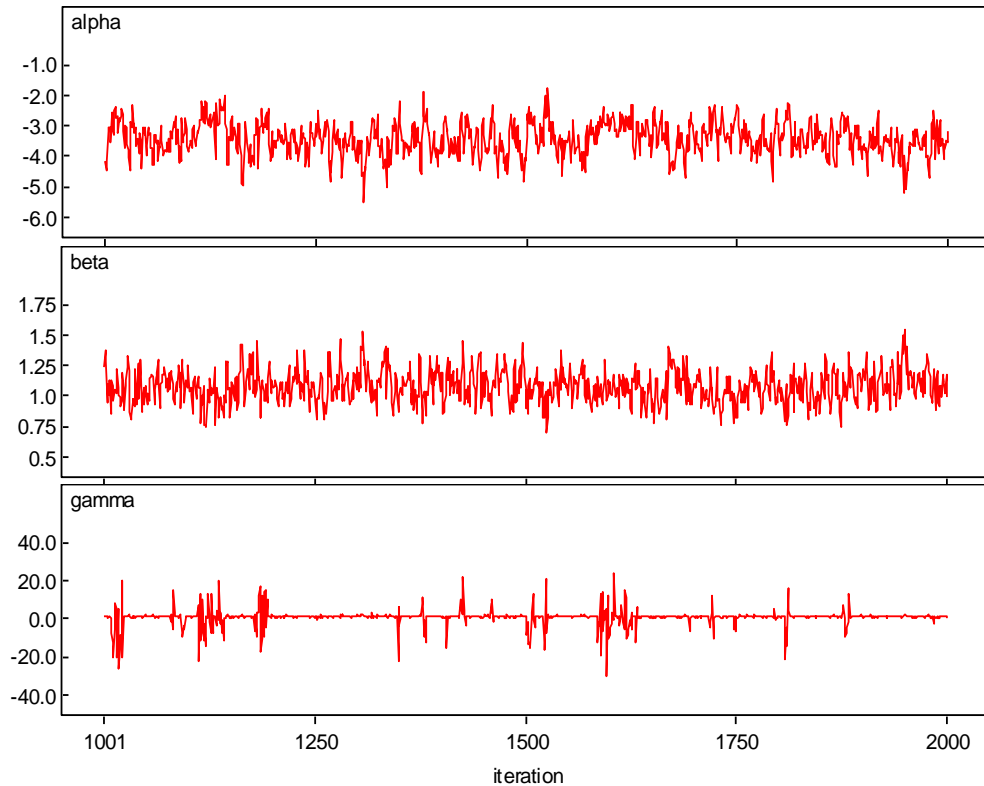
```
# -----call bugs() to fit model-----
library(R2WinBUGS)
modelName = 'd:/code/bugs/model.txt'
fit = bugs(data, inits, params, model.file = modelName,
  n.chains = 1, n.iter = 10000, n.burnin = 5000, n.thin = 5,
  bugs.seed = sample(1:9999, size=1), debug = TRUE,
  DIC = FALSE,
  bugs.directory = "d:/softwares/WinBUGS14/")
fit
fit$sims.matrix
```

## # Key WinBUGS code

```
alpha ~ dnorm(0, 0.01)
beta ~ dnorm(0, 0.01)
gamma ~ dnorm(0, 0.01)
```

```
for (i in 1:n) {
  y[i] ~ dbin(p[i], N)
  logit(p[i]) <- alpha + beta*x[i] + gamma*z[i]
}
```

# Results



node	mean	sd	MC error	2.50%	median	97.50%	start	sample
alpha	-3.423	0.5463	0.03888	-4.494	-3.406	-2.397	1001	1000
beta	1.077	0.1354	0.007451	0.827	1.07	1.364	1001	1000
gamma	1.143	1.885	0.06088	0.3462	1.187	2.05	1001	1000



# Two applications of Bayesian method in Ecology

- Occupancy model
- Hierarchical model

# Occupancy model

Need R package: R2WinBUGS (include: coda, boot)

Need to install WinBUGS14

# Species occupancy

- Conduct “presence-absence” surveys.
- Estimate the fraction of sites being occupied by a species when species is not always detected with certainty.
- The sites are arbitrarily defined spatial units (camera trap, wetland, forest patch, survey route).

**MacKenzie, D. I., et al. 2002. Estimating site occupancy rates when detection probabilities are less than one. - Ecology 83: 2248-2255.**  
Mackenzie, D. I. and Royle, J. A. 2005. Designing occupancy studies: general advice and allocating survey effort. - J. Appl. Ecol. 42: 1105-1114.  
Massolo, A. and Meriggi, A. 1998. Factors affecting habitat occupancy by wolves in northern Apennines (northern Italy): a model of habitat suitability. - Ecography 21: 97-107.  
Misenhelter, M. D. and Rotenberry, J. T. 2000. Choices and consequences of habitat occupancy and nest site selection in sage sparrows. - Ecology 81: 2892-2901.  
Peres-Neto, P. R., et al. 2001. Environmentally constrained null models: site suitability as occupancy criterion. - Oikos 93: 110-120.

# Key points of occupancy model

- Key design issues: **Replication**
  - Temporal replication: repeat visits to sample units within a relatively short period of time (e.g., an overwintering season)
  - Spatial replication: randomly selected ‘sites’ within the study area, better covering a gradient of some covariates (e.g. elevation, human impact, vegetation)

# Timing of repeated surveys

- Usually conducted as multiple discrete visits (e.g., on different days)
- Can also use multiple surveys within a single visit
  - Multiple independent observers
  - Potentially introduce heterogeneity into data
    - Single visit to each site vs. multiple visits to each site
  - Rotate observers amongst sites on each day
  - Rotate order each site is sampled within a day

# Designing occupancy surveys

- Several important issues to consider:
  1. Clear objectives that are explicitly linked to science or management
  2. Selection of sampling units
    - Probabilistic sampling design
    - Size of unit relative to species of interest
  3. Timing of repeat surveys
    - “closed”
  4. Allocation of survey effort
    - Survey all of the sites equal number of times?

# Data

- Detection history: Record for each visited site or sample unit
  - 1 denotes detection
  - 0 denotes nondetection
- Example detection history:  $h_i = 1\ 0\ 0\ 1\ 0$ 
  - Denotes 5 visits to the site, species detected during visits 1 and 4
  - 0 does not necessarily mean the species was absent

## Model Parameters: Single-Season Models

$\psi_i$  -probability site  $i$  is occupied.

$p_{ij}$  -probability of detecting the species in site  $i$  at time  $j$ , given species is present.

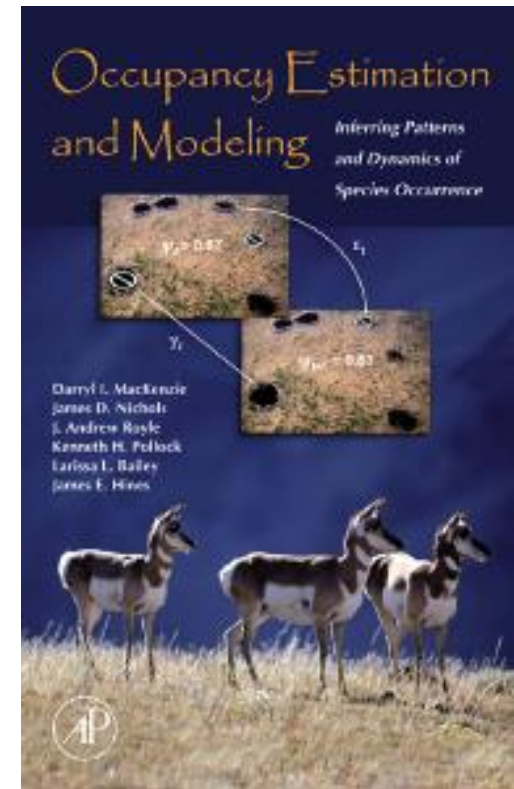


## Model assumptions

- Sites are closed to changes in occupancy state between sampling occasions
- Species are not falsely detected.
- The detection process is independent at each site  
Far enough apart to be biologically independent.
- No heterogeneity in occupancy  
Or be explained by covariates
- No heterogeneity in detection  
Or be explained by covariates

# Keys references

- D.I. MacKenzie, J.D. Nichols, J.A. Royle, K.H. Pollock, L.L. Bailey, and J.E. Hines. 2006. Occupancy estimation and modeling. Academic Press. Burlington, MA.
- D. I. MacKenzie, J. D. Nichols, J. E. Hines, M.G. Knutson, and A.B. Franklin. 2003. Estimating site occupancy, colonization, and local extinction when a species is detected imperfectly. *Ecology* 84:2200-2207.
- Barbraud, C. , J. D. Nichols, J. E. Hines, and H. Hafner. 2003. Estimating rates of local extinction and colonization in colonial species and an extension to the metapopulation and community levels. – *Oikos* 101: 113–126.



## Case study of camera trapping

237 camera-sites, mean working duration 90 days in three stages (p.1-3). At every site, elevation (elev), vegetation coverage (cover), effective monitoring area (area.1-3), the days when pictures were taken (day.1-3), were recorded. For sites without any pictures, the day.1-3 were given a median day of the stage.

CamID	p.1	p.2	p.3	elev	cover	day.1	day.2	day.3	area.1	area.2	area.3
C001	0	0	0	420	3	29	58	73	30	30	30
C002	0	1	0	450	21	13	39	62	20	19	18
C003	0	0	0	1050	32	30	47	74	15	13	11
C004	0	0	1	1110	35	23	44	71	23	21	18
C005	1	1	0	510	2	28	56	73	26	28	29
C006	0	0	1	630	60	17	56	73	19	23	24

# R code

```

setwd('d:/models/occupancy')
library("R2WinBUGS")
source("utilfns.Rd")

data      <- read.table("cam.csv", header=TRUE, sep="," , na.strings=c("NA")) # see previous slide
y         <- as.matrix(data[, c("p.1", "p.2", "p.3")])
M         <- nrow(y)
J         <- ncol(y)
cover     <- as.vector(scale(data[, "cover"], center=TRUE))
elev      <- as.vector(scale(data[, "elev"], center=TRUE))
elev2     <- elev*elev
date      <- as.matrix(data[,c("day.1", "day.2", "day.3")])
mdate     <- mean(date, na.rm=TRUE)
sddate    <- sqrt(var(date[1:length(date)], na.rm=TRUE))
date      <- (date - mdate) / sddate
date2     <- date*date

area       <- as.matrix(data[,c("area.1", "area.2", "area.3")])
mint       <- mean(area, na.rm=TRUE)
sdint      <- sqrt(var(area[1:length(area)], na.rm=TRUE))
intensity  <- (area-mint)/sdint

date[is.na(y)] <- date2[is.na(y)] <- intensity[is.na(y)] <- 0

data      <- list ( "y", "M", "J", "cover", "elev", "elev2", "date", "date2", "intensity")
inits     <- function()
  list ( z=rbinom(M,1,.4), psi0=runif(1), b1=rnorm(1), b2=rnorm(1),
        b3=rnorm(1), p0=runif(1), , a2=rnorm(1), a3=rnorm(1))
parameters <- c("b0", "b1", "b2", "b3", "a0", "a1", "a2", "a3", "p0", "psi0")

```

"utilfns.Rd"

```

`expit` <- function(x){
  exp(x)/(1+exp(x))
}

```

```

`logit` <- function(x){
  log(x/(1-x))
}

```

## R code

```

sink("model.txt")
cat("
model {
# prior distributions
p0 ~ dunif(0,1)
a0 <- log(p0/(1-p0))
a1 ~ dnorm(0,.001)
a2 ~ dnorm(0,.001)
a3 ~ dnorm(0,.001)
psi0 ~ dunif(0,1)
b0 <- log(psi0/(1-psi0))
b1 ~ dnorm(0,.001)
b2 ~ dnorm(0,.001)
b3 ~ dnorm(0,.001)
for(i in 1:M){
  z[i] ~ dbin(psi[i],1) # STATE MODEL
  logit(psi[i]) <- b0 + b1*elev[i] + b2*elev2[i] + b3*cover[i]
  for(t in 1:J){
    mu[i, t] <- a0 + a1*date[i, t] + a2*date2[i, t] + a3*intensity[i, t]
    p[i, t] <- exp(mu[i, t])/(1+exp(mu[i, t]))
    muy[i, t] <- z[i]*p[i, t]
    y[i, t] ~ dbin(muy[i, t],1) # OBSERVATION MODEL
  }
}
}
",fill=TRUE)
sink()

```

```

fit = bugs(data, inits, parameters, "model.txt",
          n.chains=3, n.iter=2000, n.burnin=1000, n.thin=1,
          bugs.seed=sample(1:9999, size=1),
          debug=TRUE, DIC=FALSE,
          bugs.directory = "d:/softwares/WinBUGS14/")
fit

```

# Models in WinBUGS format

```
for(i in 1:M){ # M sites
```

```
  z[i] ~ dbin(psi[i],1) # STATE MODEL, psi[i] is the probability of occurrence at site i
```

```
  logit(psi[i]) <- b0 + b1*elev[i] + b2*elev2[i] + b3*forest[i] # probability of occ.
```

```
for(t in 1:J){ # J is the number of surveys at site i, which is 3
```

```
  mu[i, t] <- a0 + a1*date[i, t] + a2*date2[i, t] + a3*intensity[i, t]
```

```
  p[i, t] <- exp(mu[i, t])/(1+exp(mu[i, t])) # probability of detection
```

```
  mui[i, t] <- z[i]*p[i, t] # realized detection
```

```
  y[i, t] ~ dbin(mui[i, t],1) # OBSERVATION MODEL
```

```
}
```

```
}
```

# Results

node	mean	sd	MC error	2.50%	median	97.50%	start	sample
a0	1.288	0.236	0.01057	0.814	1.293	1.752	1001	3000
a1	0.1816	0.1775	0.008539	-0.1515	0.1811	0.5331	1001	3000
a2	-0.04908	0.1576	0.008315	-0.3486	-0.04592	0.2685	1001	3000
a3	0.1571	0.2308	0.01171	-0.2552	0.1324	0.6192	1001	3000
b0	-0.1762	0.2751	0.01053	-0.7205	-0.1768	0.3636	1001	3000
b1	2.043	0.3083	0.01126	1.468	2.035	2.691	1001	3000
b2	-1.156	0.2654	0.01152	-1.716	-1.143	-0.6635	1001	3000
b3	0.8612	0.2362	0.007432	0.4179	0.8551	1.357	1001	3000
p0	0.7812	0.04028	0.001818	0.693	0.7846	0.8522	1001	3000
psi0	0.4568	0.06704	0.002553	0.3273	0.4559	0.5899	1001	3000

# Hierarchical modeling

## An example

Abundance of a species at site  $i$ :  $N_i = \text{Poisson}(\lambda_i)$

Detection rate of an individual at site  $i$ :  $r_i$

Parameters to be estimated:  $\lambda_i$   $r_i$



# History of hierarchical modeling

The idea of hierarchical modeling started in the mid 20th century (Gelman et al. 1995). The hierarchical spatio-temporal dynamic model methodology was illustrated with a case study concerned with predicting the abundance of the house finch (*Carpodacus mexicanus*) over the eastern half of the U.S. from 1966 through 2001, with data collected during the North American Breeding Bird Survey (BBS; Robbins et al. 1986).

Hierarchical models in environmental sciences were developed in recent studies (Berliner 1996; Wikle et al. 1998, 2001).

Hierarchical modeling is based on the simple fact from probability that the joint distribution of a collection of random variables can be decomposed into a series of conditional models (Wikle 2003a, 2003b).

Berliner, L. M. 1996. Hierarchical Bayesian time series models.

Gelman, A., Carlin, J. B., Stern, H. S., and Rubin, D. B. 1995. Bayesian Data Analysis (1st ed.), London: Chapman and Hall.

Wikle, C. K. 2003a. Hierarchical Bayesian models for predicting the spread of ecological processes. *Ecology* **84**:1382-1394.

Wikle, C. K. 2003b. Hierarchical models in environmental science. *International Statistical Review* **71**:181-199.

Wikle, C. K., L. M. Berliner, and N. Cressie. 1998. Hierarchical Bayesian space-time models. *Environmental and Ecological Statistics* **5**:117-154.

Wikle, C. K., R. F. Milliff, D. Nychka, and L. M. Berliner. 2001. Spatiotemporal hierarchical Bayesian modeling: Tropical ocean surface winds. *Journal of the American Statistical Association* **96**:382-397.

# Key literatures

All Publications > [Ecology](#) > June 2003 > HIERARCHICAL BAYESIAN MODELS FOR PREDICTING THE SPREAD OF ECOLOGICAL P... [Advanced Search](#)

Volume 84, Issue 6 (June)

< [Previous](#) [Next](#) >

[Add to Favorites](#) | [Email](#) | [Download to Citation Manager](#) | [Track Citations](#) | [Permissions](#)

[Full-text](#) [PDF](#)

Wikle, Christopher K. 2003. HIERARCHICAL BAYESIAN MODELS FOR PREDICTING THE SPREAD OF ECOLOGICAL PROCESSES. *Ecology* 84:1382–1394. [doi:10.1890/0012-9658(2003)084[1382:HBMFPTJ]2.0.CO;2]

Special Feature

**HIERARCHICAL BAYESIAN MODELS FOR PREDICTING THE SPREAD OF ECOLOGICAL PROCESSES**

Christopher K. Wikle<sup>1</sup>

Department of Statistics, 222 Math Science Building, University of Missouri, Columbia, Missouri 65211 USA

There is increasing interest in predicting ecological processes. Methods to accomplish such predictions must account for uncertainties in observation, sampling, models, and parameters. Statistical methods for spatiotemporal processes are powerful, yet difficult to implement in complicated high-dimensional settings. However, recent advances in hierarchical formulations for such processes can be utilized for ecological prediction. These formulations are able to account for the various sources of uncertainty and can incorporate scientific judgment in a probabilistically consistent manner. In particular, analytical diffusion models can serve as motivation for the hierarchical model for invasive species. We demonstrate by example that such a framework can be utilized to predict, spatially and temporally, the relative population abundance of House Finches over the eastern United States.

Corresponding Editor (ad hoc): J. S. Clark.

Keywords: [Bayesian models](#), [Carpodacus mexicanus](#), [diffusion](#), [forecast](#), [hierarchical models](#), [House Finch](#), [invasive species](#), [state space](#), [uncertainty](#)

Received: December 3, 2001; Revised: May 7, 2002; Accepted: May 23, 2002; Final version received: June 28, 2002

<sup>1</sup> E-mail: [wikle@stat.missouri.edu](mailto:wikle@stat.missouri.edu)

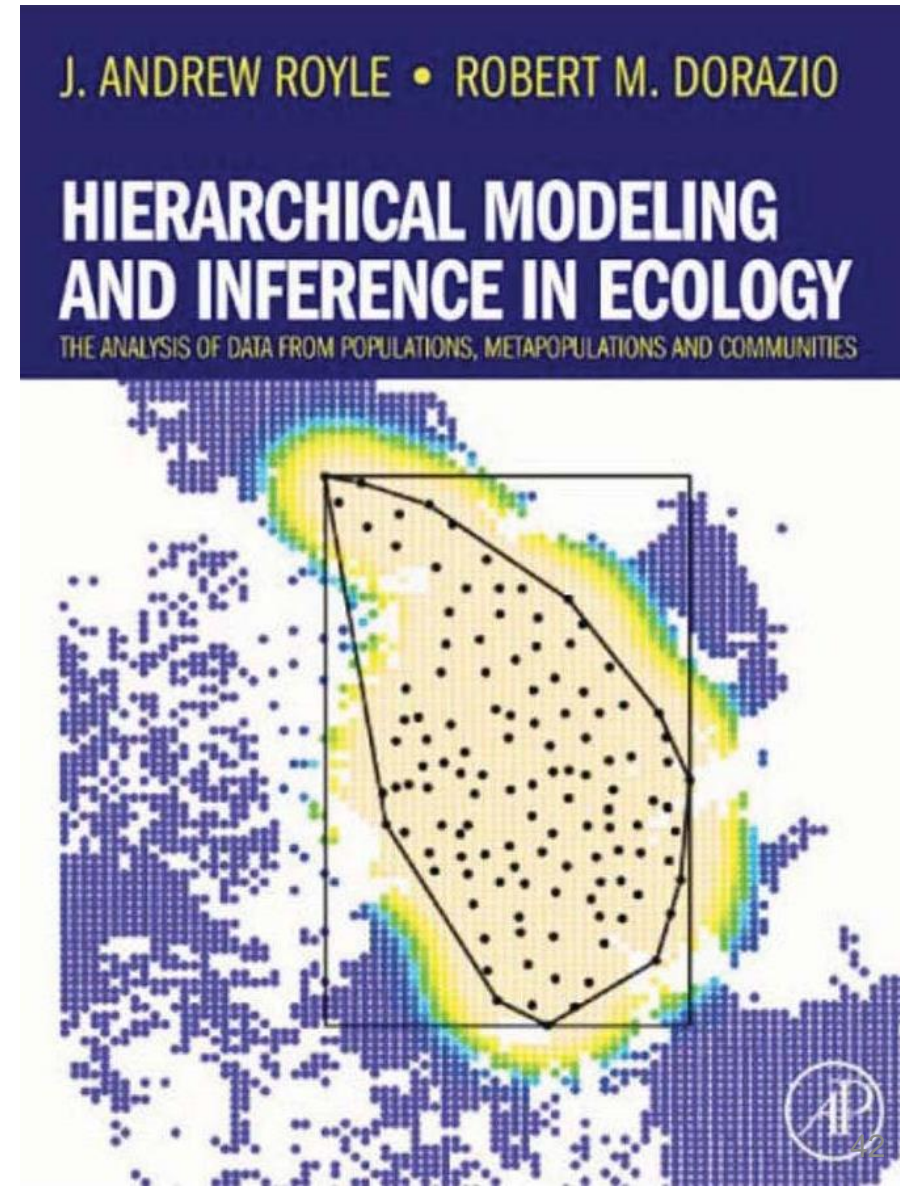
Journal Information

ISSN: 0012-9658  
Frequency: Monthly

[Mission and Scope](#)  
[Types of contributions](#)  
[Editorial Board](#)  
[Staff](#)  
[Instructions for Authors](#)  
[Reviewer Guidelines](#)

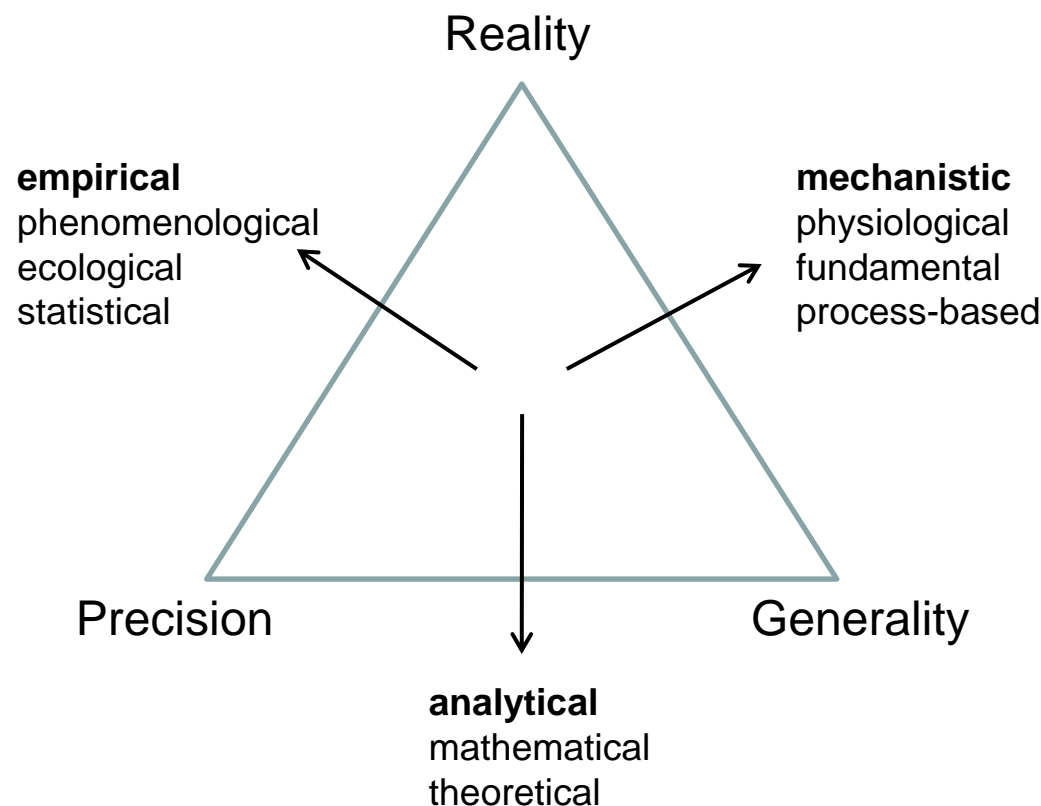
Wikle, C. K. 2003. Hierarchical Bayesian models for predicting the spread of ecological processes. *Ecology* **84**:1382–1394.

Royle, J. and R. Dorazio. 2008. Hierarchical Modeling and Inference in Ecology: The Analysis of Data from Populations, Metapopulations and Communities. Academic Press.

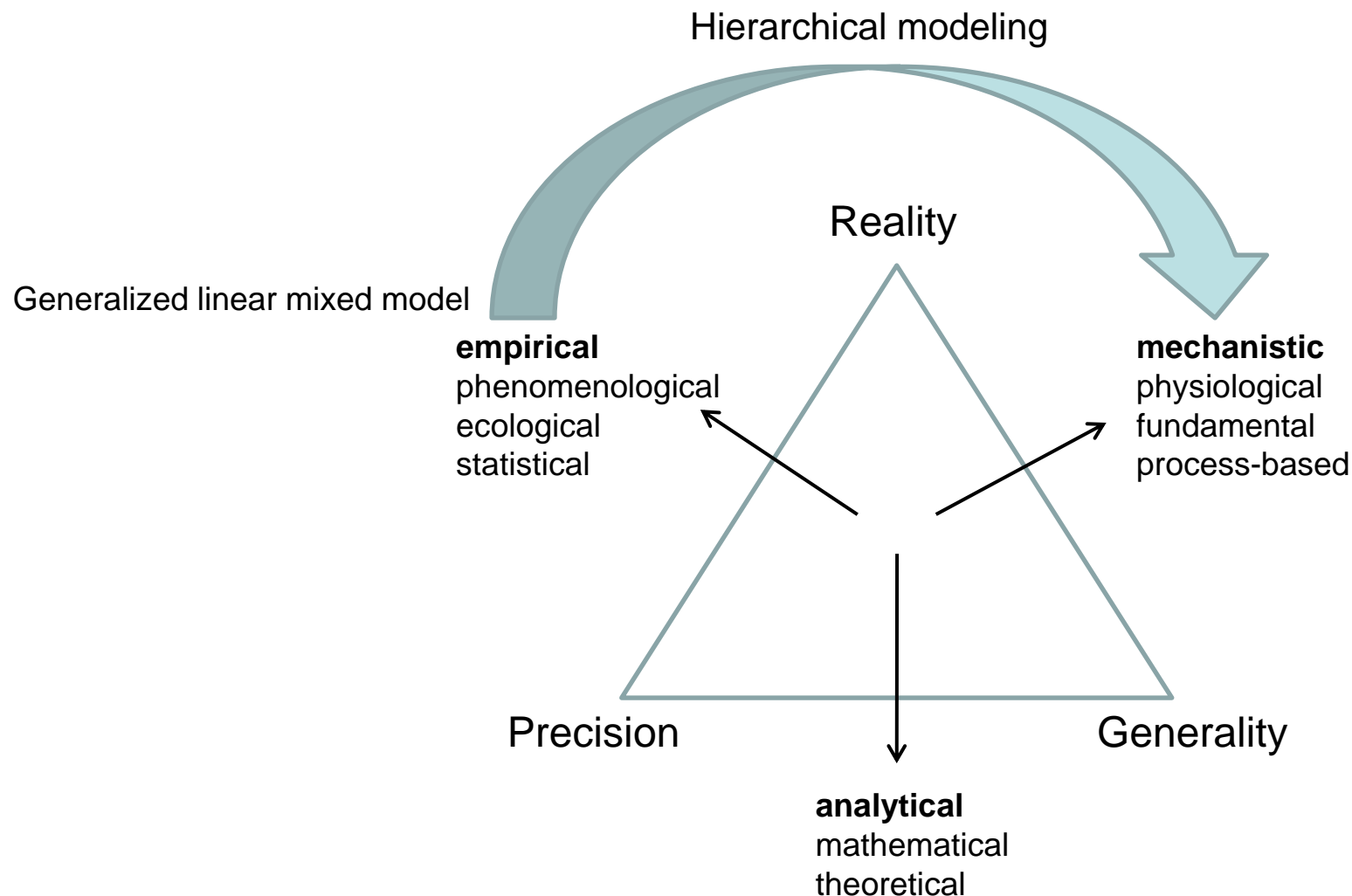


# A classification of models based on their intrinsic properties

After Levins (1966), Sharpe (1990), and Guisan & Zimmermann (2000)



# Improvement from hierarchical modeling



# An example of hierarchical modeling

(Royle and Dorazio 2008 Page 88)

Swiss bird survey data consisting of 3 replicate quadrat counts of the willow tit (*Parus montanus*, 褐头山雀) during the breeding season and covariates elevation (m) and forest cover (percent). Only a subset of quadrat counts are shown here. The symbol 'NA' indicates a missing value in an R data set.

rep1	rep2	rep3	elevation	forest
0	0	0	910	21
0	0	0	540	11
0	1	NA	1880	32
1	0	1	1400	32
0	1	1	1210	75
1	0	0	380	23
0	0	NA	1840	0
1	0	NA	1910	18
1	2	2	1630	33
0	0	0	1540	21
2	4	3	1340	39
2	3	1	1410	52
14	13	9	2030	36
6	6	8	1880	66

# Models in WinBUGS format

```
for(i in 1:M){ # M sites
```

```
  z[i] ~ dbin(psi[i],1) # STATE MODEL, psi[i] is the probability of occurrence at site i
```

```
  logit(psi[i]) <- b0 + b1*elev[i] + b2*elev2[i] + b3*forest[i] # probability of occ.
```

```
for(t in 1:J){ # J is the number of surveys at site i, which is 3
```

```
  mu[i, t] <- a0 + a1*date[i, t] + a2*date2[i, t] + a3*intensity[i, t]
```

```
  p[i, t] <- exp(mu[i, t])/(1+exp(mu[i, t])) # probability of detection
```

```
  muy[i, t] <- z[i]*p[i, t] # realized detection
```

```
  y[i, t] ~ dbin(muy[i, t],1) # OBSERVATION MODEL
```

```
}
```

```
}
```

# R code

```

setwd('d:/models/hierarchical_modeling')
library("R2WinBUGS")
source("utilfns.Rd")

data      <- read.table("wtmatrix.csv", header=TRUE, sep=",", na.strings=c("NA")) # see previous slide
y         <- as.matrix(data[, c("y.1", "y.2", "y.3")])
M         <- nrow(y)
J         <- ncol(y)
length    <- as.vector(data[, "length"])
forest    <- as.vector(scale(data[, "forest"], center=TRUE))
elev      <- as.vector(scale(data[, "elev"], center=TRUE))
elev2     <- elev*elev
date      <- as.matrix(data[,c("day.1", "day.2", "day.3")])
mdate     <- mean(date, na.rm=TRUE)
sddate    <- sqrt(var(date[1:length(date)], na.rm=TRUE))
date      <- (date - mdate) / sddate
date2     <- date*date
dur       <- as.matrix(data[, c("dur.1", "dur.2", "dur.3")])
intensity <- dur/length
mint      <- mean(intensity, na.rm=TRUE)
sdint     <- sqrt(var(intensity[1:length(intensity)], na.rm=TRUE))
intensity <- (intensity - mint)/sdint
length    <- (length - mean(length))/sqrt(var(length))
date[is.na(y)] <- date2[is.na(y)] <- intensity[is.na(y)] <- 0

data      <- list ( "y", "M", "J", "forest", "elev", "elev2", "date", "date2", "intensity")
inits     <- function()
  list ( z=rbinom(M,1,.4),psi0=runif(1),b1=rnorm(1),b2=rnorm(1),
        b3=rnorm(1),p0=runif(1),a1=rnorm(1),a2=rnorm(1),a3=rnorm(1))
parameters <- c("b0", "b1", "b2", "b3", "a0", "a1", "a2", "a3", "p0", "psi0")

```

## "utilfns.Rd"

```

`expit` <- function(x){
  exp(x)/(1+exp(x))
}

`logit` <- function(x){
  log(x/(1-x))
}

```

## R code

```

sink("model.txt")
cat("
model {
# prior distributions
p0      ~ dunif(0,1)
a0      <- log(p0/(1-p0))
a1      ~ dnorm(0,.001)
a2      ~ dnorm(0,.001)
a3      ~ dnorm(0,.001)
psi0    ~ dunif(0,1)
b0      <- log(psi0/(1-psi0))
b1      ~ dnorm(0,.001)
b2      ~ dnorm(0,.001)
b3      ~ dnorm(0,.001)
for(i in 1:M){
  z[i] ~ dbin(psi[i],1) # STATE MODEL
  logit(psi[i]) <- b0 + b1*elev[i] + b2*elev2[i] + b3*forest[i]
  for(t in 1:J){
    mu[i, t] <- a0 + a1*date[i, t] + a2*date2[i, t] + a3*intensity[i, t]
    p[i, t] <- exp(mu[i, t])/(1+exp(mu[i, t]))
    muy[i, t] <- z[i]*p[i, t]
    y[i, t] ~ dbin(muy[i, t],1) # OBSERVATION MODEL
  }
}
}
", fill = TRUE)
sink()

```

```

fit = bugs(data, inits, parameters, "model.txt",
          n.chains=3, n.iter=2000, n.burnin=1000, n.thin=1,
          bugs.seed=sample(1:9999, size=1),
          debug=TRUE, DIC=FALSE,
          bugs.directory = "d:/softwares/WinBUGS14/")
fit

```



# Results

node	mean	sd	MC error	2.50%	median	97.50%	start	sample
a0	1.288	0.236	0.01057	0.814	1.293	1.752	1001	3000
a1	0.1816	0.1775	0.008539	-0.1515	0.1811	0.5331	1001	3000
a2	-0.04908	0.1576	0.008315	-0.3486	-0.04592	0.2685	1001	3000
a3	0.1571	0.2308	0.01171	-0.2552	0.1324	0.6192	1001	3000
b0	-0.1762	0.2751	0.01053	-0.7205	-0.1768	0.3636	1001	3000
b1	2.043	0.3083	0.01126	1.468	2.035	2.691	1001	3000
b2	-1.156	0.2654	0.01152	-1.716	-1.143	-0.6635	1001	3000
b3	0.8612	0.2362	0.007432	0.4179	0.8551	1.357	1001	3000
p0	0.7812	0.04028	0.001818	0.693	0.7846	0.8522	1001	3000
psi0	0.4568	0.06704	0.002553	0.3273	0.4559	0.5899	1001	3000



# Fitting model to the willow tit data using the likelihood specification (non-Bayesian method)

(Royle and Dorazio 2008 Page 117)

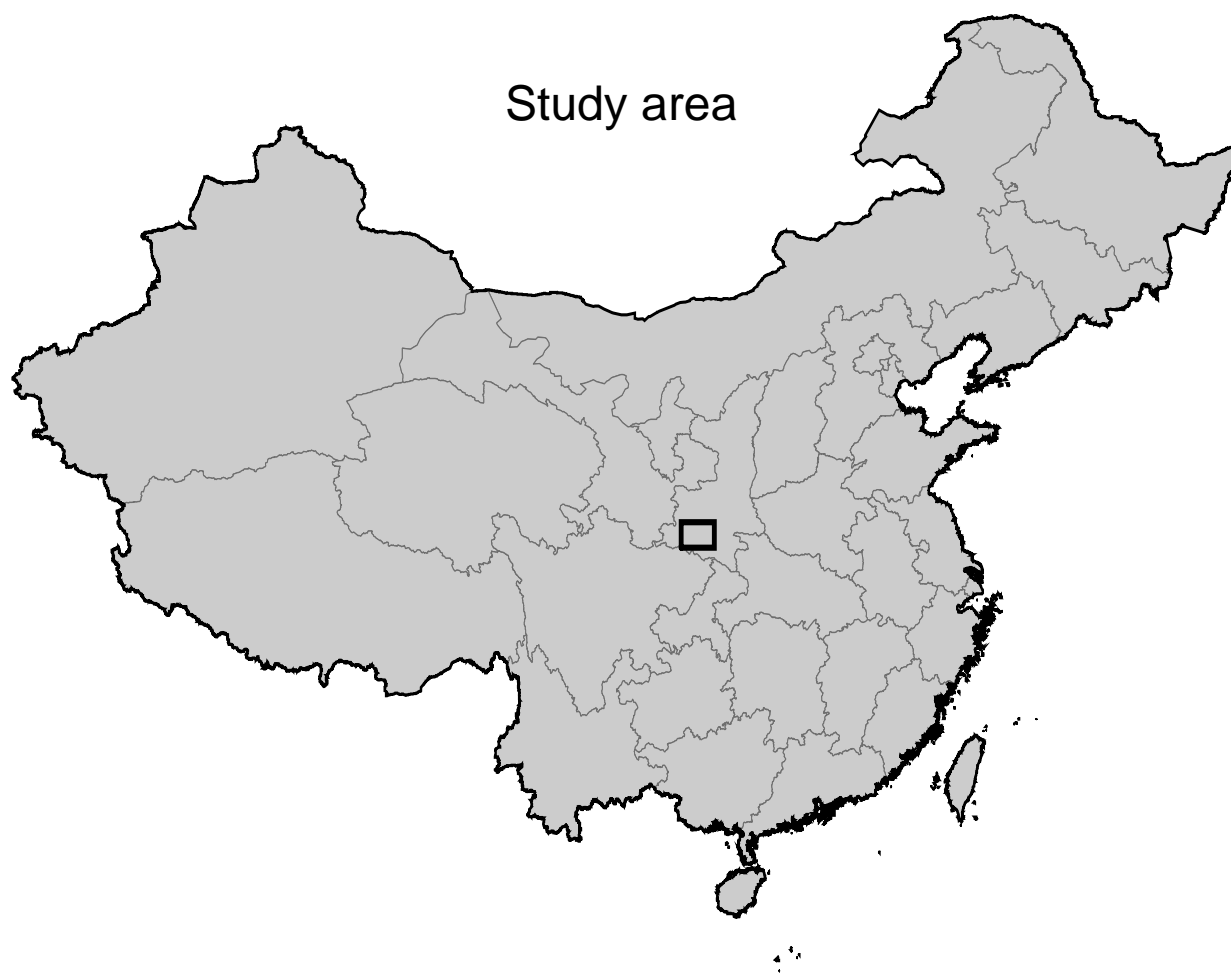
```
# Data
source("D:/models/hierarchical_modeling/utlfn.Rd")
data <- read.table("wtmatrix.csv", header=TRUE, sep="," , na.strings=c("NA"))
forest <- as.vector(scale(data[, "forest"], center=TRUE))
elev <- as.vector(scale(data[, "elev"], center=TRUE))
ymat <- as.matrix(data[, c("y.1", "y.2", "y.3")])
M <- nrow(ymat)
length <- as.vector(data[, "length"])
date <- as.matrix(data[, c("day.1", "day.2", "day.3")])
mdate <- mean(date, na.rm=TRUE)
sddate <- sqrt(var(date[1:length(date)], na.rm=TRUE))
date <- (date - mdate) / sddate
dur <- as.matrix(data[, c("dur.1", "dur.2", "dur.3")])
intensity <- dur/length
mint <- mean(intensity, na.rm=TRUE)
sdint <- sqrt(var(intensity[1:length(intensity)], na.rm=TRUE))
intensity <- (intensity - mint)/sdint
length <- (length - mean(length))/sqrt(var(length))
```

## # Function for parameter estimation

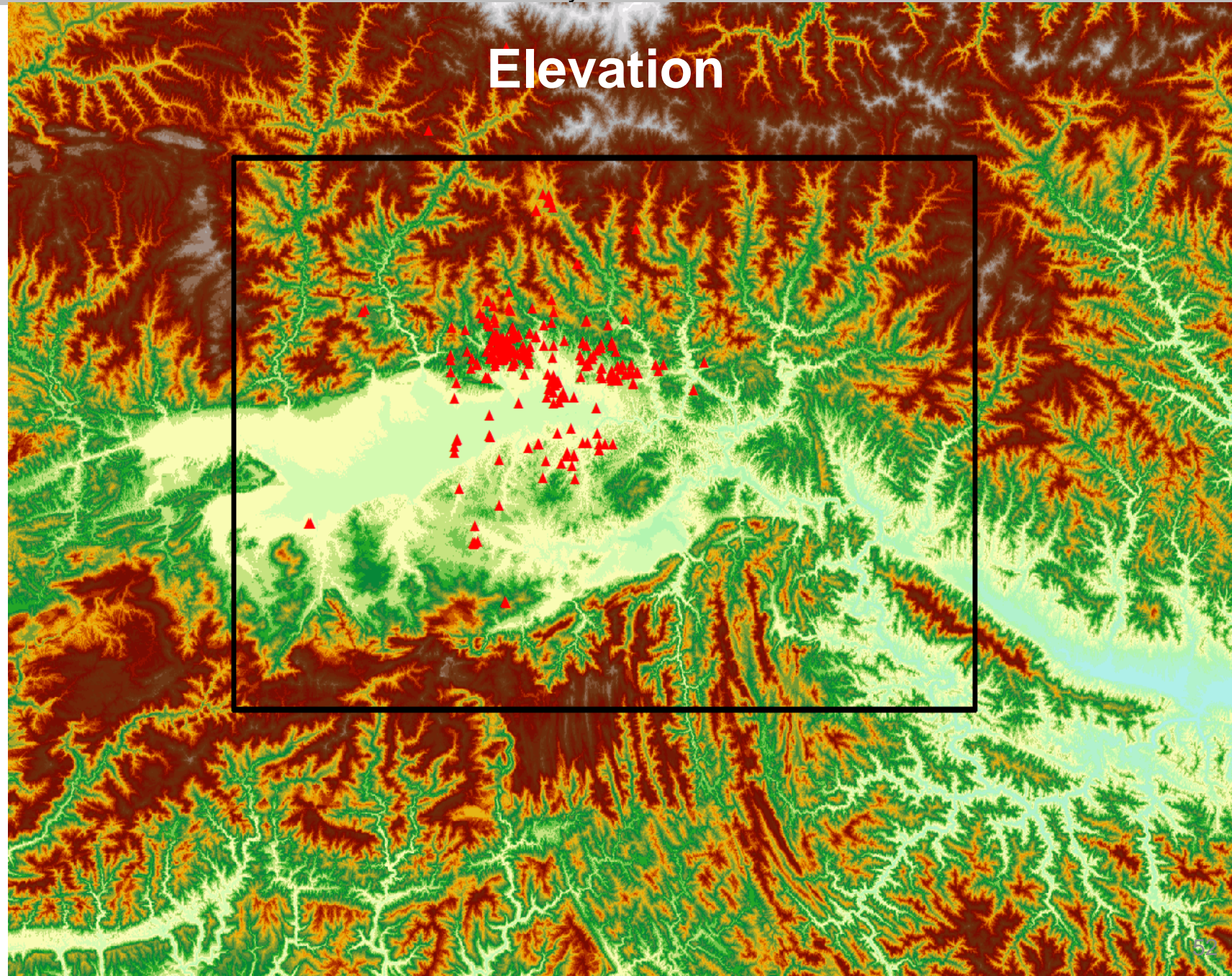
```
lik <- function(parms,vars){
  tmp <- rep(0,9)
  names(tmp) <- c("pconst","psiconst","length","elev1","elev2",
                 "forest","intensity","date1","date2")

  tmp[vars] <- parms
  ones <- rep(1,M) # number of row
  pmat <- expit (tmp[1]*ones + tmp[7]*intensity + tmp[8]*date
                + tmp[9]*(date^2)) # detection rate
  psi <- expit (tmp[2]*ones + tmp[3]*length + tmp[4]*elev
               + tmp[5]*(elev^2) + tmp[6]*forest) # occurrences rate
  loglik <- rep(NA, M)
  for(i in 1:M){
    yvec <- ymat[i,]
    navec <- is.na(yvec)
    nd <- sum(yvec[!navec])
    pvec <- pmat[i,]
    cp <- (pvec^yvec)*((1-pvec)^(1-yvec))
    cp[navec] <- 1
    loglik[i] <- log(prod(cp)*psi[i] + ifelse(nd==0,1,0)*(1-psi[i]))
  }
  sum(-1*loglik)
}
```

# Case study – nest site selection of crested ibis

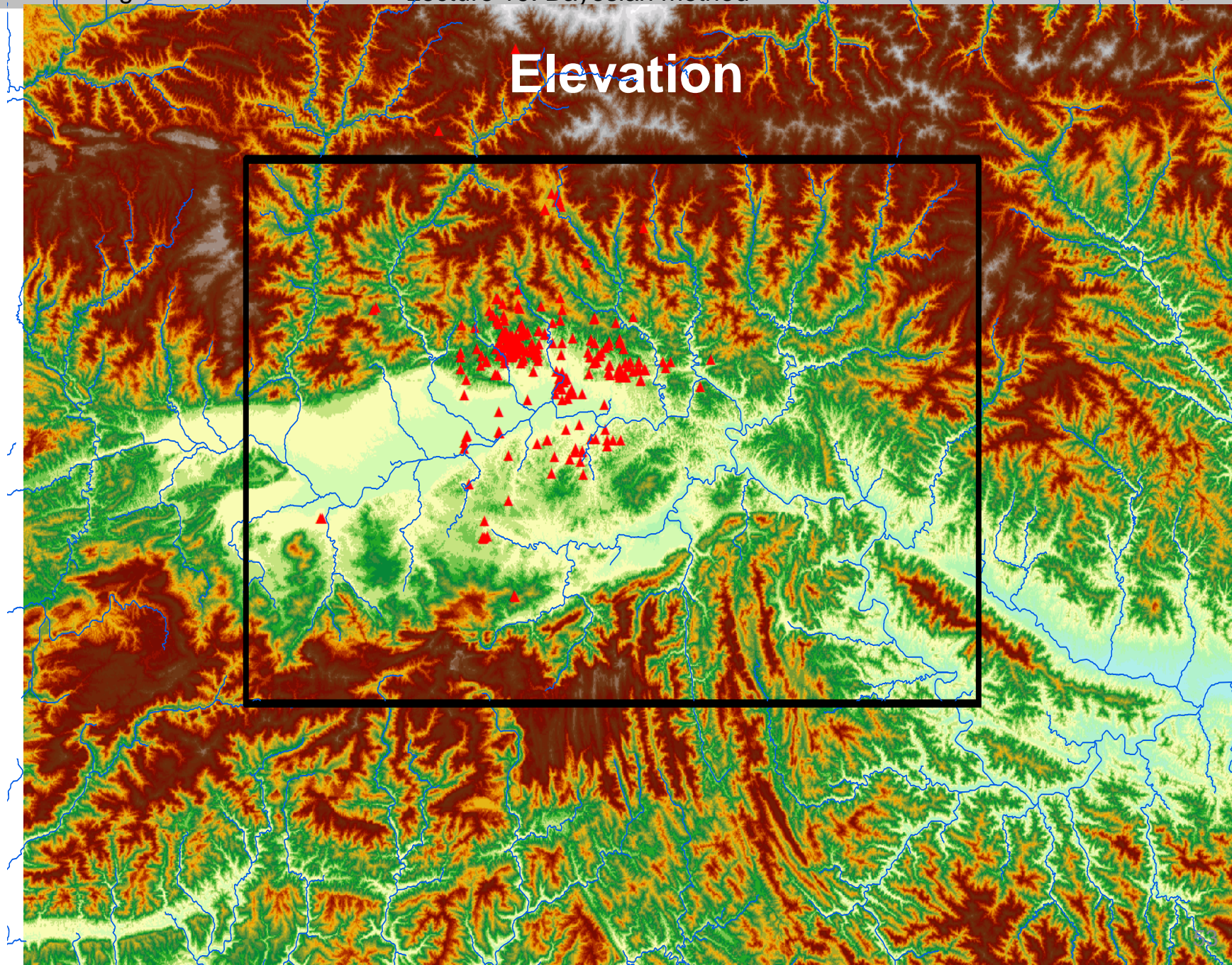






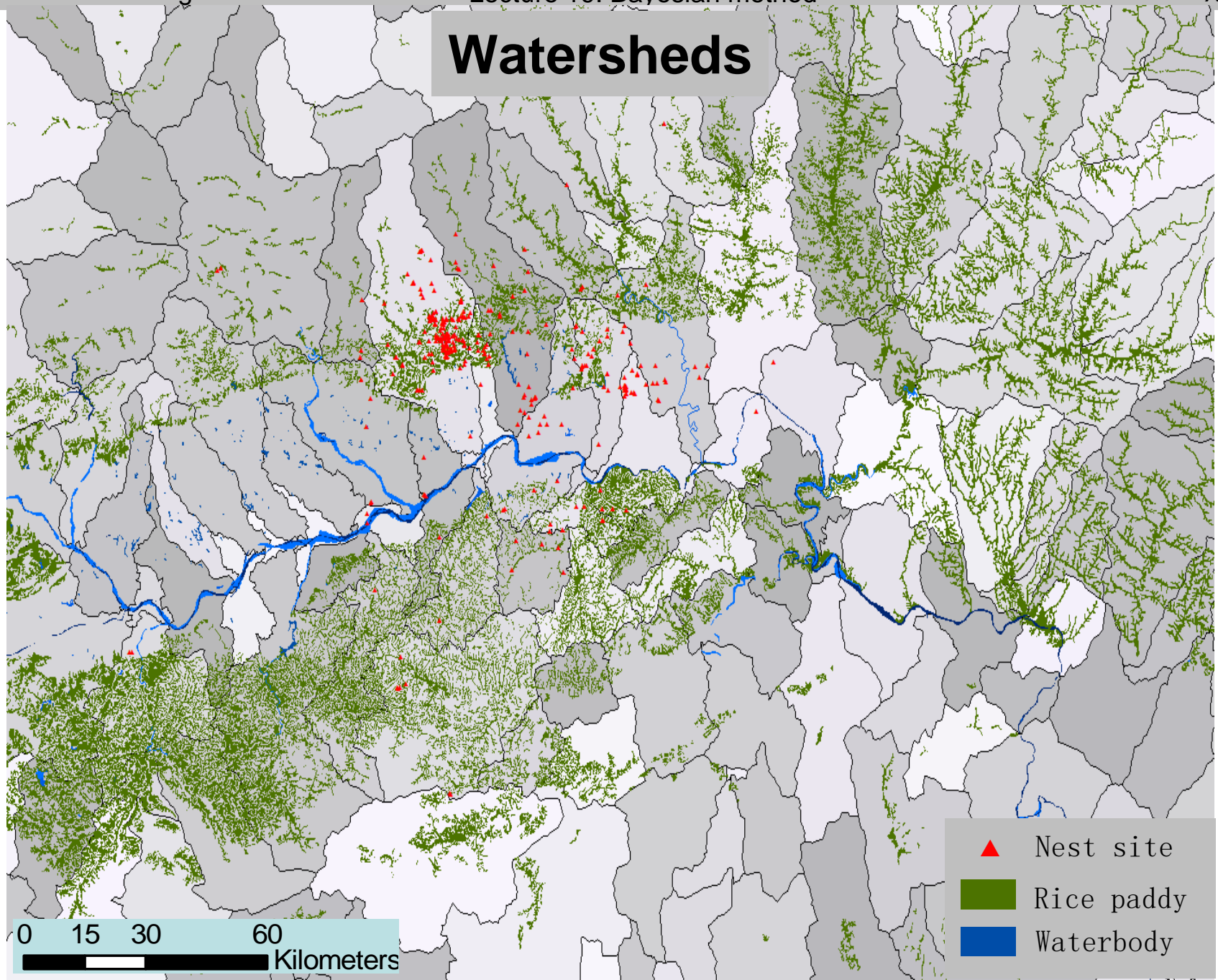


# Elevation





# Watersheds

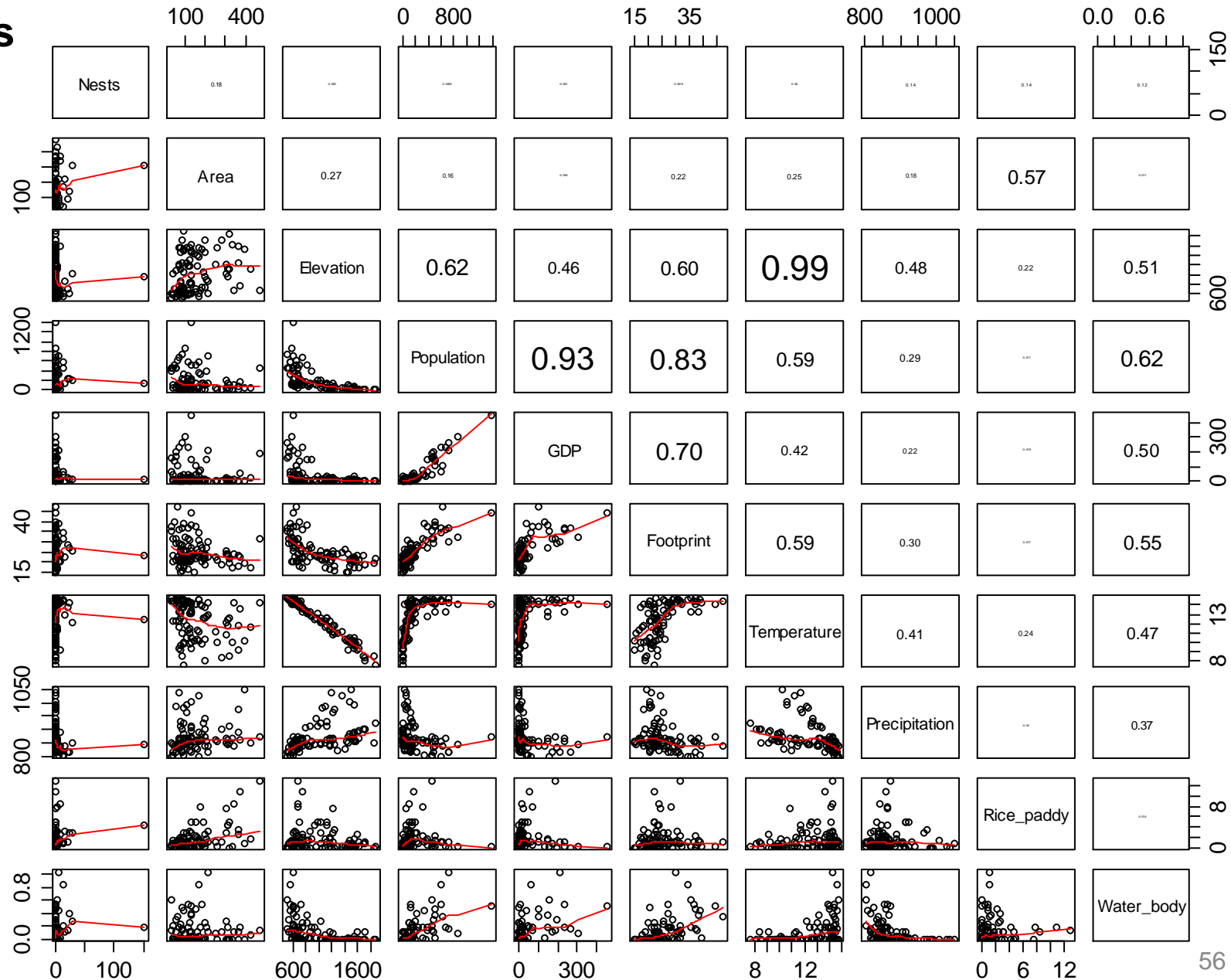


# Variables

## In every watersheds

- Average elevation
- SD of elevation
- Area of rice paddy
- Area of water body
- Human footprint index
- Population density
- GDP
- Temperature
- Precipitation
- Area of the watershed

# Correlation of variables





# Variable standardization

```
sheds <- read.csv('d:/text/ibis_Bayes/watersheds4.csv', header=TRUE) #read data

elev <- as.vector(scale(sheds[, "elev"], center=TRUE)) #elevation
elev.sd <- as.vector(scale(sheds[, "elev.SD"], center=TRUE)) #variance of elevation
foot <- as.vector(scale(sheds[, "footprint"], center=TRUE)) #Human Footprint Index
foot2 <- foot*foot #The square term
elev2 <- elev*elev #The square term
precip <- as.vector(scale(sheds[, "precip"], center=TRUE)) #Precipitation
T <- as.vector(scale(sheds[, "T"], center=TRUE)) #Temperature
pop <- as.vector(scale(sheds[, "pop"], center=TRUE)) #human population density
paddy <- as.vector(scale(sheds[, "paddy"], center=TRUE)) #area of rice paddy
water <- as.vector(scale(sheds[, "water"], center=TRUE)) #area of water body (pond and river)
wetland <- as.vector(scale(sheds[, "paddy"] * sheds[, "water"], center=TRUE)) #interaction term
```

Variable **wetland** is the product of **paddy** and **waterbody**.

# Variable selection using Generalized additive model (GAM)

```
library(mgcv)
model = gam(log(count + 1) ~ s(x1) + s(x2) + ... + s(x10), data = sheds)

Xs <- c("elev", "elev.SD", "footprint", "GDP", "population", "precip",
        "paddy", "waterbody", "wetland", "temperature")
```

From the full model, select models with 2-5 variables

Based on: deviance explained and GCV (Generalized Cross Validation)

# WinBUGS code

```

library(R2WinBUGS)
# Write WinBUGS code
sink("d:/text/ibis/ibis.txt")
cat("
model{
  for(i in 1:nsites){
    logit(p[i])    <- a0 + a1*foot[i] + a2*foot2[i]    #detection rate
    logit(psi[i]) <- b0 + b1*wet[i]  + b2*precip[i]   #rate of occurrence
    z[i] ~ dbin(psi[i],1)      #1=presence, 0=absence
    tmp[i] <- z[i]*p[i]        #rate of recording
    y[i] ~ dbin(tmp[i],J[i])   #number of recording
  }
  a0 ~ dnorm(0,.001)
  a1 ~ dnorm(0,.001)
  a2 ~ dnorm(0,.001)
  b0 ~ dnorm(0,.001)
  b1 ~ dnorm(0,.001)
  b2 ~ dnorm(0,.001)
  logit(psi0) <- b0 #截距对应的 p值要指定,或者直接给初始值
  logit(p0)    <- a0 #同上
}
",fill=TRUE)
sink()

data <- list ( "y","J","nsites","foot","foot2","wet","precip")
inits <- function()
  list(z=rbinom(nsites,1,.4),a0=rnorm(1),a1=rnorm(1),a2=rnorm(1),
       b0=rnorm(1),b1=rnorm(1),b2=rnorm(1))
parameters <- c("a0","a1","a2","b0","b1","b2","psi0","p0")

out <- bugs(data, inits, parameters, "d:/text/ibis/ibis.txt",
  n.chain=3, n.burnin=1000, n.iter=10000, debug=T,
  bugs.directory = "d:/softwares/WinBUGS14/")

```

## WinBUGS code

# The probability of selecting watershed i as nest site:

```
logit(psi[i]) <- b0 + b1*wetland[i] + b2*precipitation[i]
```

# The detection rate of a nest in watershed i:

```
logit(p[i]) <- a0 + a1*footprint[i] + a2*footprint2[i]
```

# Whether or not the crested ibis selected watershed i as nest site:

```
z[i] ~ dbin(psi[i], 1)    #1=presence, 0=absence
```

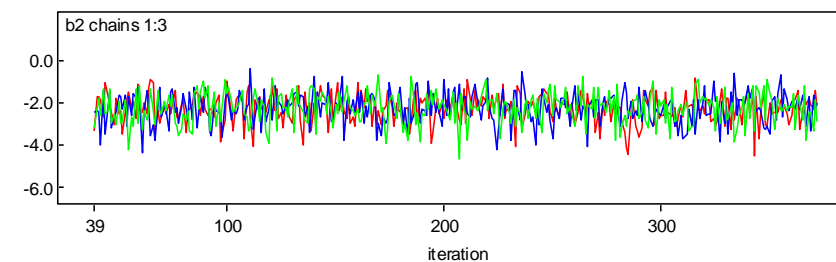
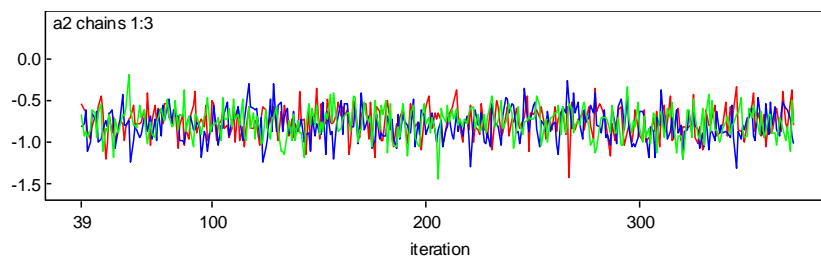
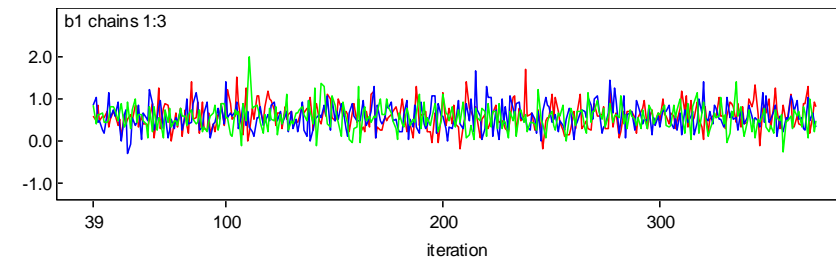
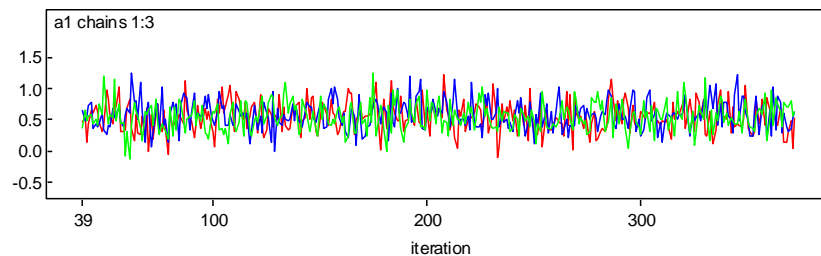
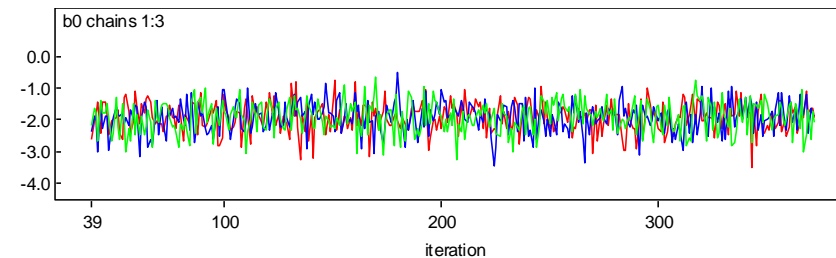
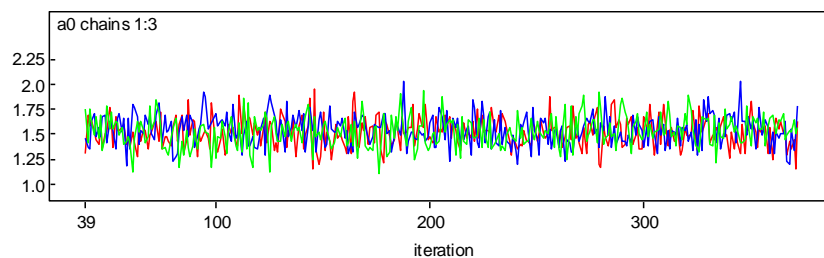
# The probability of recording a nest (nest existed and was found) in watershed i:

```
tmp[i] <- z[i]*p[i]
```

# The number of nests recorded in watershed i. (J[i] is the number of surveys in watershed i):

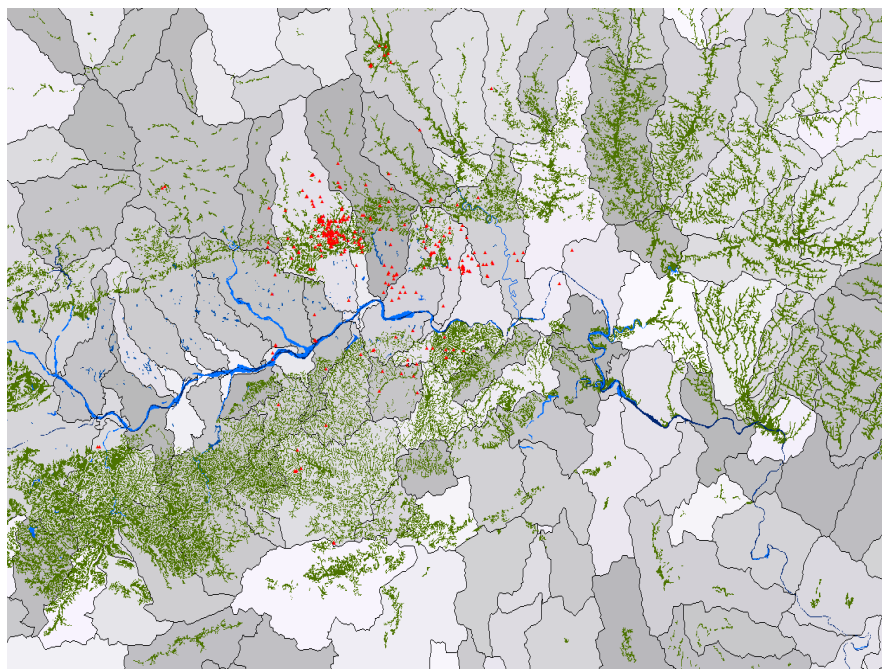
```
y[i] ~ dbin(tmp[i], J[i])
```

# Parameter estimation using MCMC



# Parameter estimation

node	mean	sd	MC error	2.50%	median	97.50%
a0	1.531	0.1484	0.005556	1.256	1.533	1.836
a1_footprint	0.5567	0.2303	0.00751	0.129	0.5492	1.029
a2_footprint <sup>2</sup>	-0.7765	0.18	0.006249	-1.129	-0.7768	-0.4362
b0	-1.924	0.4737	0.01423	-2.897	-1.893	-1.062
b1_wetland	0.5695	0.295	0.008542	0.02534	0.5557	1.207
b2_precipitation	-2.272	0.701	0.02063	-3.78	-2.222	-1.033



# Assignment

General objectives: learn Bayesian method.

- Develop a dataset to perform:
  - Bayesian method  $Y - X_1, X_2, X_3, \text{ etc.}$
- Describe the dataset. Give the assumed prior distribution and calculated posterior distribution of the parameters. Make a conclusion.