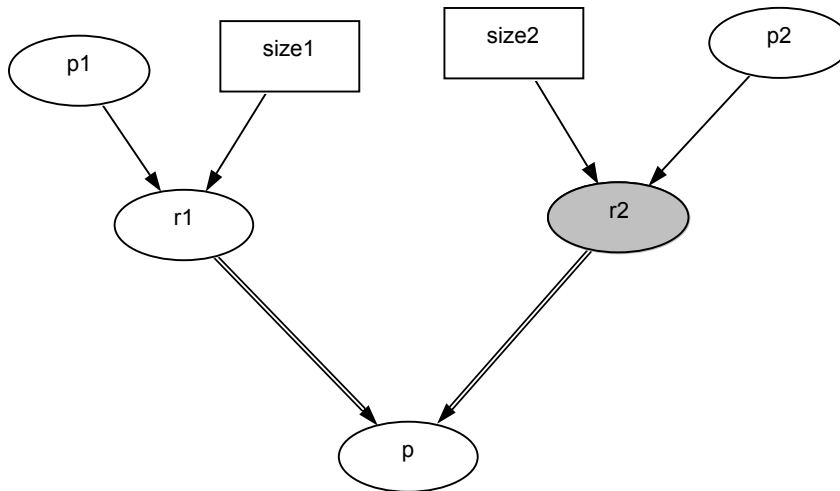


1 EXAMPLE 1 (BASHIR)

```
model;
{
  r1 ~ dbin(p1,size1)
  r2 ~ dbin(p2,size2)
  p <- (r1 + r2) / (size1 + size2)
  p1 ~ dnorm(0.25,38416)
  p2 ~ dnorm(0.35,38416)
}

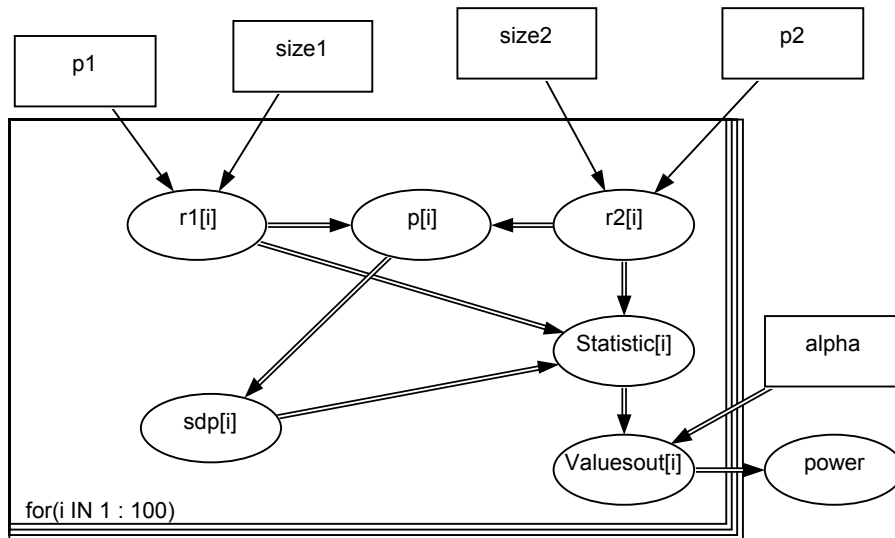
list(size1=250, size2=150) # data
list(p1=0.25,p2=0.35)    # initial values
```

name:	r2	type:	stochastic	density:	dbin
proportion	p2	order	size2	lower bound	upper bound



2 EXAMPLE 2 (BASHIR)

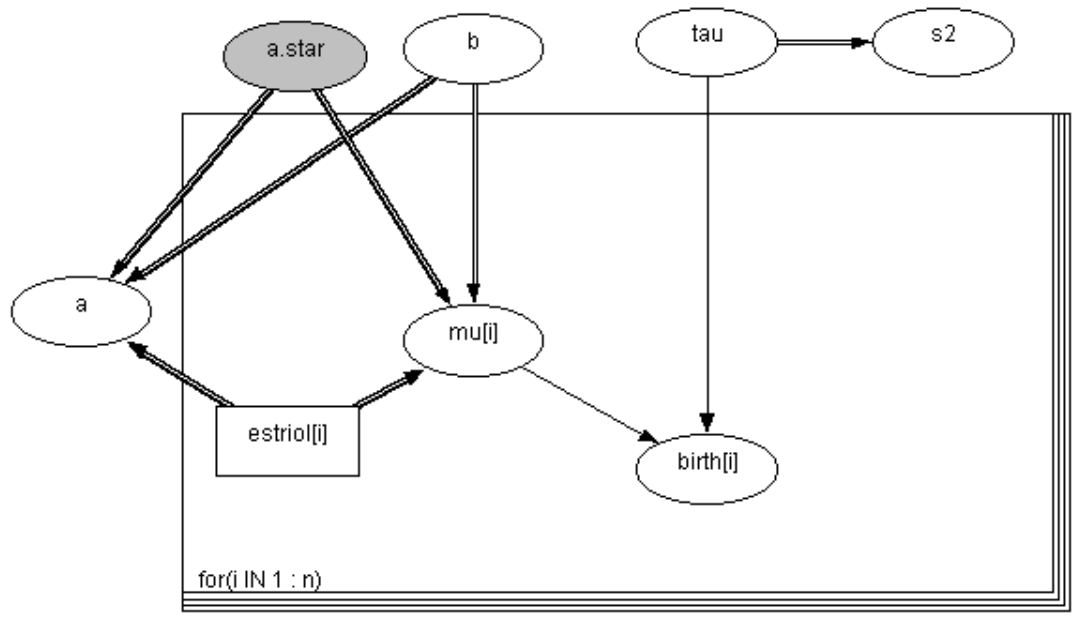
```
list(size1=250,size2=150, p1=0.25,p2=0.35,alpha=1.96) # data
model;
{
  for( i in 1 : 100 ) {
    r1[i] ~ dbin(p1,size1)
  }
  for( i in 1 : 100 ) {
    r2[i] ~ dbin(p2,size2)
  }
  for( i in 1 : 100 ) {
    p[i] <- (r1[i] + r2[i]) / (size1 + size2)
  }
  for( i in 1 : 100 ) {
    sdp[i] <- sqrt((p[i] * (1 - p[i])) * (1 / size1 + 1 / size2))
  }
  for( i in 1 : 100 ) {
    Statistic[i] <- (r1[i] / size1 - r2[i] / size2) / sdp[i]
  }
  for( i in 1 : 100 ) {
    Valuesout[i] <- step(abs(Statistic[i]) - alpha)
  }
  power <- mean(Valuesout[])
}
```



3 EXAMPLE 1: BIRTHWEIGHT & ESTRIOL LEVEL

```
{
#   definition of likelihood function
#
  for (i in 1:n) {
    birth[i]~dnorm( mu[i], tau ); # random component
    mu[i]<-a.star+b*(estriol[i]-mean(estriol[])); # systematic
component
                                     #   & link function
  }
#   prior distributions
#
  a.star~dnorm( 0, 1.0E-04 ); # normal prior for a
  b~dnorm( 0, 1.0E-04 ); # normal prior for b
  tau~dgamma( 1.0E-04 , 1.0E-04 ); # gamma prior for precision
  s2<-1/tau;
  a<-a.star-b*mean(estriol[]);
}
list(a.star=0.0, b=0.0, tau=1.0) # initial values

# data
list(n=31)
estriol[] birth[]
7      25
9      25
9      25
12     27
14     27
16     27
16     24
14     30
16     30
16     31
17     30
19     31
21     30
24     28
15     32
16     32
17     32
25     32
27     34
15     34
15     34
15     35
16     35
19     34
18     35
17     36
18     37
20     38
22     40
25     39
24     43
```



4 Example 2: BEETLES DATASET

```
model beetles;
{
  for (i in 1:N) {
    r[i] ~ dbin(p[i], n[i]);
    logit(p[i]) <- alpha.star + beta*(x[i]-mean(x[]));
    r.hat[i] <- p[i]*n[i]; # fitted values
  }
  alpha.star ~ dnorm(0.0, 1.0E-3);
  beta ~ dnorm(0.0, 1.0E-3);
  alpha <- alpha.star - beta*mean(x[]);
  odds.ratio <- exp( beta )
}
```

```
list(x = c(1.6907, 1.7242, 1.7552, 1.7842, 1.8113, 1.8369, 1.8610, 1.8839) n = c(59, 60, 62, 56, 63, 59, 62, 60) r =
c(6, 13, 18, 28, 52, 53, 61, 60), N=8) #data
list(alpha.star=0, beta=0) #initial values
```

