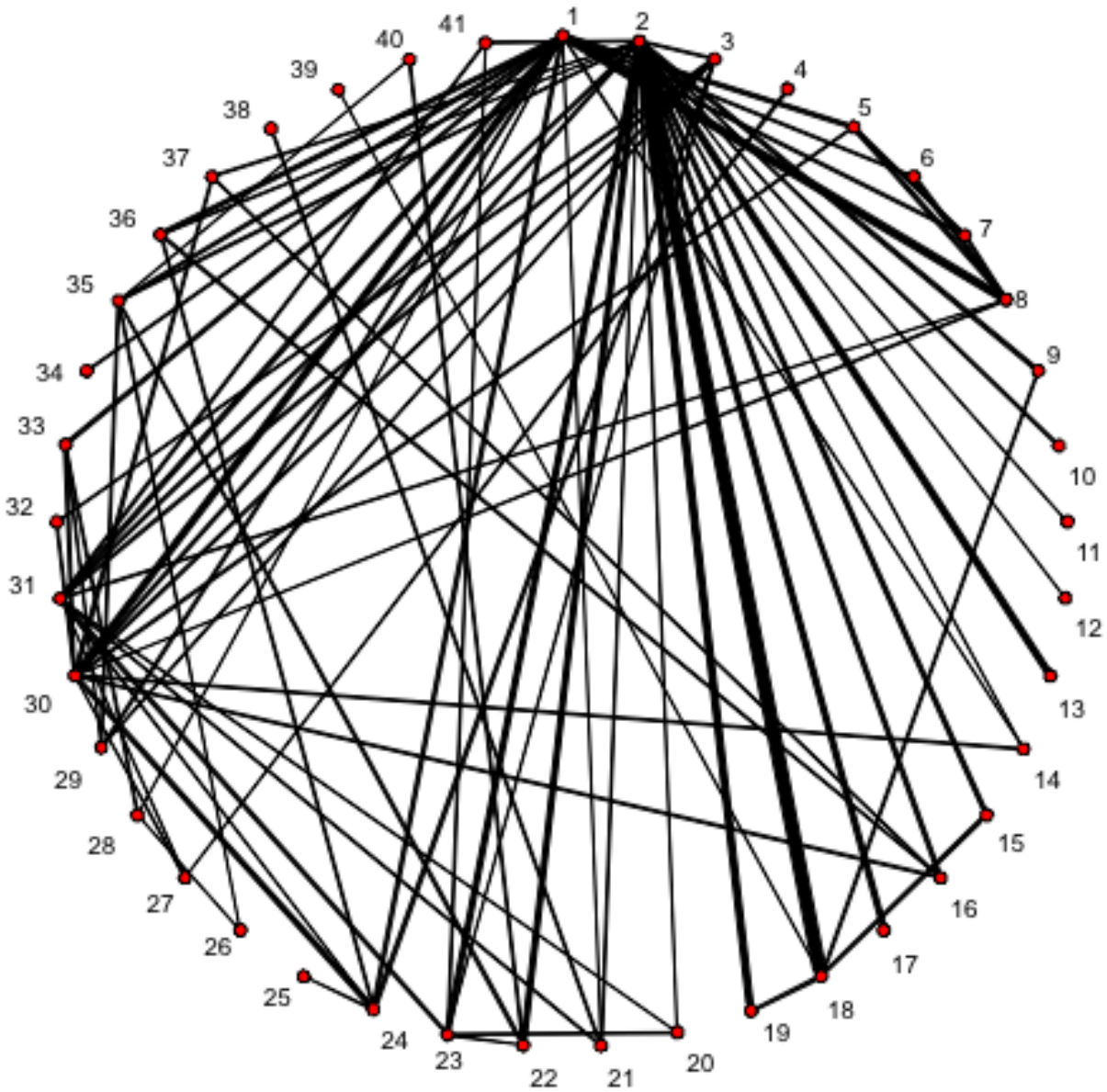
APPENDIX 11: NETWORK META-ANALYSIS DIAGRAMS AND WINBUGS CODE

Symptoms of social anxiety disorder diagram 2
Recovery diagram 3
Symptom versus recovery relationship 4
WinBUGS code 5

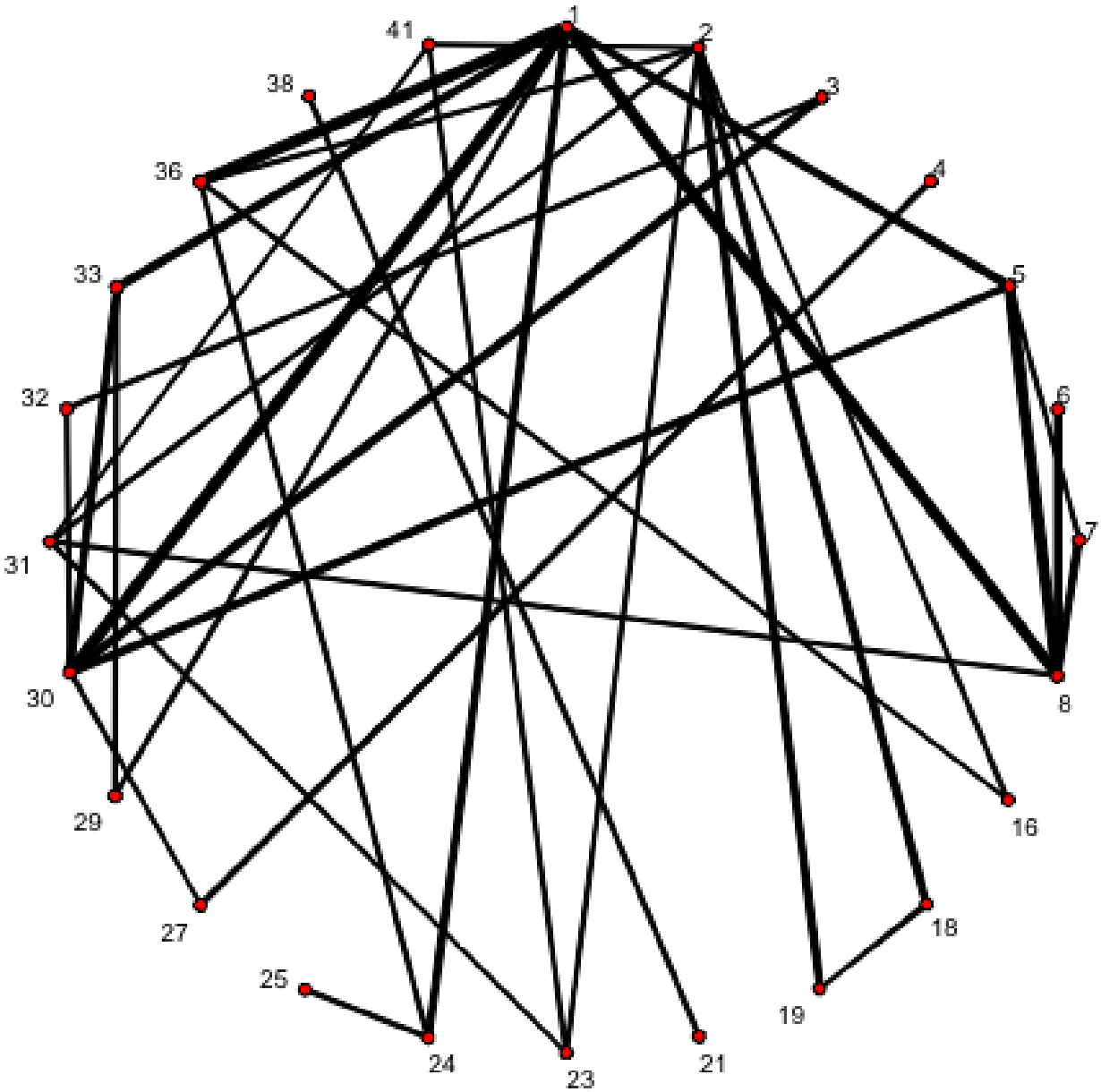
Abbreviations

LOR	log-odds ratio
SMD	standardised mean difference

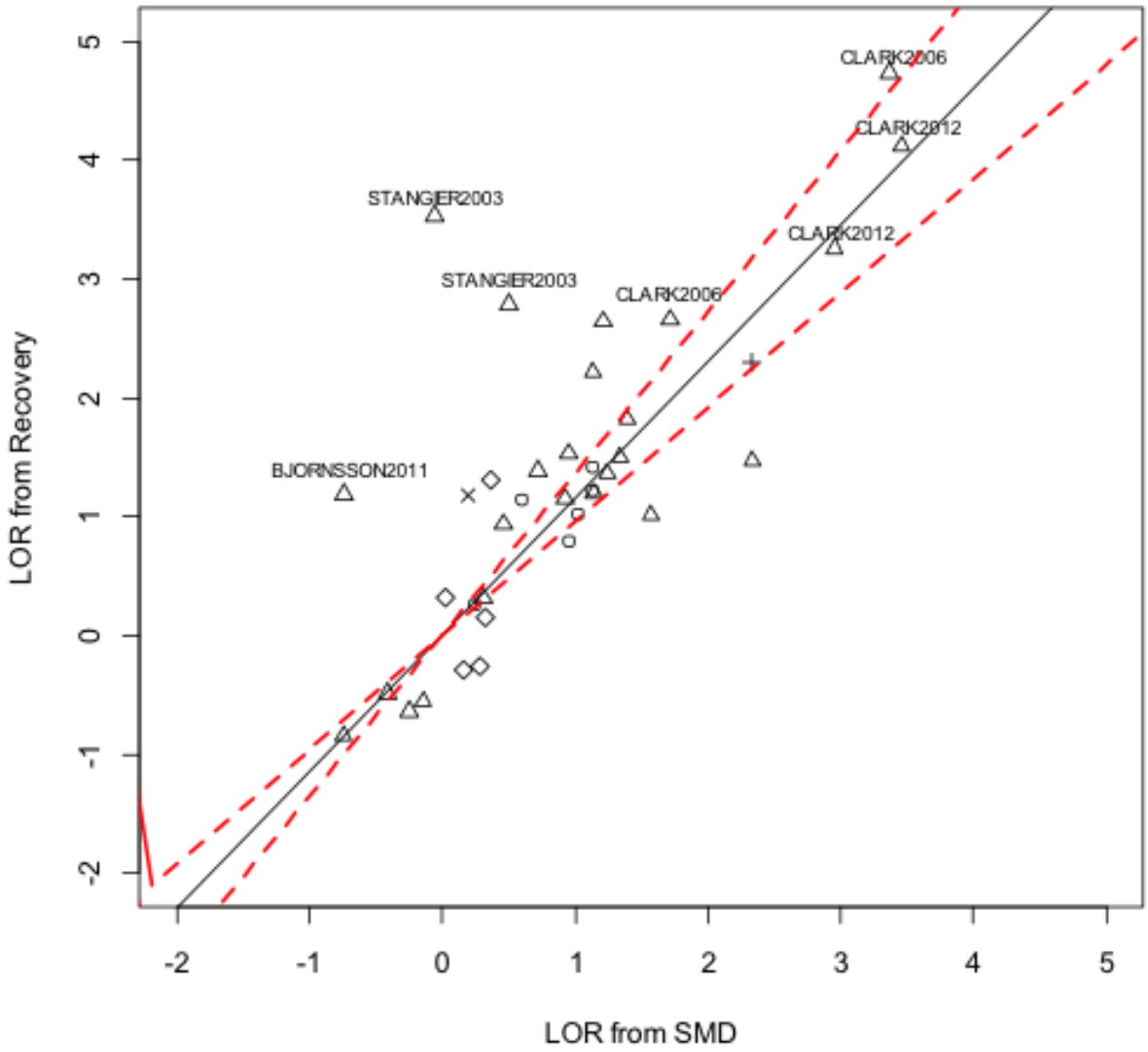
Symptoms of social anxiety disorder diagram



Recovery diagram



Symptom versus recovery relationship



WinBUGS code

```
# Normal likelihood, identity link for trial-level data given as SMD
# Class model: Random effects model for multi-arm trials
# Includes calibration regression for Recovery
model{                          # *** PROGRAM STARTS
# 2 ARM STUDIES SMD
for(i in 1:ns2) {               # LOOP THROUGH 2-ARM STUDIES
   prec[i,2] <- 1/Var[i,2]      # set precisions
   y[i,2] ~ dnorm(delta[i,2],prec[i,2]) # normal likelihood for 2-arm trials
#Deviance contribution for trial i
   resdev[i] <- (y[i,2]-delta[i,2])*(y[i,2]-delta[i,2])*prec[i,2]
  }
# 3 ARM STUDIES SMD
for(i in 1:ns3) {               # LOOP THROUGH THREE-ARM STUDIES
   for (k in 1:(na[i+ns2]-1)) { # set variance-covariance matrix
      for (j in 1:(na[i+ns2]-1)) {
         Sigma[i,j,k] <- V[i+ns2]*(1-equals(j,k)) + Var[i+ns2,k+1]*equals(j,k)
        }
     }
# Precision matrix
   Omega[i,1:(na[i+ns2]-1),1:(na[i+ns2]-1)] <- inverse(Sigma[i,,])
# multivariate normal likelihood for 3-arm trials
   y[i+ns2,2:na[i+ns2]] ~ dmnorm(delta[i+ns2,2:na[i+ns2]],Omega[i,1:2,1:2])
# Deviance contribution for trial i
   for (k in 1:(na[i+ns2]-1)){  # multiply vector & matrix
      ydiff[i,k]<- y[i+ns2,(k+1)] - delta[i+ns2,(k+1)]
      z[i,k]<- inprod2(Omega[i,k,1:2], ydiff[i,1:2])
                  }
   resdev[i+ns2]<- inprod2(ydiff[i,1:(na[i+ns2]-1)], z[i,1:(na[i+ns2]-1)])
  }
# 4 ARM STUDIES SMD
for(i in 1:ns4) {               # LOOP THROUGH FOUR-ARM STUDIES
  for (k in 1:(na[i+ns2+ns3]-1)) {  # set variance-covariance matrix
   for (j in 1:(na[i+ns2+ns3]-1)) {
     Sigma4[i,j,k] <- V[i+ns2+ns3]*(1-equals(j,k)) +
Var[i+ns2+ns3,k+1]*equals(j,k)
     }
  }
# Precision matrix
   Omega4[i,1:(na[i+ns2+ns3]-1),1:(na[i+ns2+ns3]-1)] <- inverse(Sigma4[i,,])
# multivariate normal likelihood for 4-arm trials
   y[i+ns2+ns3,2:na[i+ns2+ns3]] ~
dmnorm(delta[i+ns2+ns3,2:na[i+ns2+ns3]],Omega4[i,1:3,1:3])
#Deviance contribution for trial i
   for (k in 1:(na[i+ns2+ns3]-1)){  # multiply vector & matrix
```

```
      ydiff4[i,k]<- y[i+ns2+ns3,(k+1)] - delta[i+ns2+ns3,(k+1)]
      z4[i,k]<- inprod2(Omega4[i,k,1:3], ydiff4[i,1:(na[i+ns2+ns3]-1)])
                  }
   resdev[i+ns2+ns3]<- inprod2(ydiff4[i,1:(na[i+ns2+ns3]-1)],
z4[i,1:(na[i+ns2+ns3]-1)])
  }
# 5 ARM STUDIES SMD
for(i in 1:ns5) {             # LOOP THROUGH FIVE-ARM STUDIES
# set variance-covariance matrix
  for (k in 1:(na[i+ns-ns5]-1)) {  # set variance-covariance matrix
   for (j in 1:(na[i+ns-ns5]-1)) {
     Sigma5[i,j,k] <- V[i+ns-ns5]*(1-equals(j,k)) + Var[i+ns-ns5,k+1]*equals(j,k)
     }
  }
# Precision matrix
   Omega5[i,1:(na[i+ns-ns5]-1),1:(na[i+ns-ns5]-1)] <- inverse(Sigma5[i,,])
# multivariate normal likelihood for 5-arm trials
   y[i+ns-ns5,2:na[i+ns-ns5]] ~ dmnorm(delta[i+ns-ns5,2:na[i+ns-
ns5]],Omega5[i,1:4,1:4])
# Deviance contribution for trial i
   for (k in 1:(na[i+ns-ns5]-1)){  # multiply vector & matrix
      ydiff5[i,k]<- y[i+ns-ns5,(k+1)] - delta[i+ns-ns5,(k+1)]
      z5[i,k]<- inprod2(Omega5[i,k,1:4], ydiff5[i,1:(na[i+ns-ns5]-1)])
                  }
   resdev[i+ns-ns5]<- inprod2(ydiff5[i,1:(na[i+ns-ns5]-1)], z5[i,1:(na[i+ns-ns5]-
1)])
  }
# ALL STUDIES WITH SMD, RE MODEL
for(i in 1:ns){               #   LOOP THROUGH ALL STUDIES
   w[i,1] <- 0    # adjustment for multi-arm trials is zero for control arm
   delta[i,1] <- 0            # treatment effect is zero for control arm
   for (k in 2:na[i]) {        # LOOP THROUGH ARMS
# trial-specific LOR distributions
      delta[i,k] ~ dnorm(md[i,k],taud[i,k])
# mean of random effects distributions, with multi-arm trial correction
      md[i,k] <-  d[t[i,k]] - d[t[i,1]] + sw[i,k]
# precision of random effects distributions (with multi-arm trial correction)
      taud[i,k] <- tau *2*(k-1)/k
# adjustment, multi-arm RCTs
      w[i,k] <- (delta[i,k] - d[t[i,k]] + d[t[i,1]])
# cumulative adjustment for multi-arm trials
      sw[i,k] <- sum(w[i,1:k-1])/(k-1)
     }
  }
totresdev <- sum(resdev[])      # Total Residual Deviance
#
```

```
d[1] <- 0      # treatment effect (SMD) is zero for reference treatment
# own class treatments with zero variance
d[2] ~ dnorm(0, .0001)      # vague prior for treatment effects (SMD)
d[3] ~ dnorm(0, .0001)      # vague prior for treatment effects (SMD)
d[29] ~ dnorm(0, .0001)      # vague prior for treatment effects (SMD)
# own class treatments borrowing variance
d[4] ~ dnorm(m[D[4]], prec2[5])    # variance from class E=5
d[12] ~ dnorm(m[D[12]], prec2[9])    # variance from class I=9
# Class model (informative prior for within-class precision)
for (k in 5:11){
   d[k] ~ dnorm(m[D[k]], prec2[D[k]]) # treatment effects from Class (SMD)
  }
for (k in 13:28){
   d[k] ~ dnorm(m[D[k]], prec2[D[k]]) # treatment effects from Class (SMD)
  }
for (k in 30:nt){
   d[k] ~ dnorm(m[D[k]], prec2[D[k]]) # treatment effects from Class (SMD)
  }
m[1] <- 0
m[2] <- d[2]      # no class effect
m[3] <- d[3]      # no class effect
m[14] <- d[29]    # no class effect
for (k in 4:13){
   m[k] ~ dnorm(0, .0001)      # prior for mean class effect
   prec2[k] ~ dgamma(a,b)      # prior for class precision
   sd2[k] <- pow(prec2[k], -0.5)
}
for (k in 15:nc){
   m[k] ~ dnorm(0, .0001)      # prior for mean class effect
   prec2[k] ~ dgamma(a,b)      # prior for class precision
   sd2[k] <- pow(prec2[k], -0.5)
}
#
sd ~ dunif(0,5)              # vague prior for between-trial SD
tau <- pow(sd,-2)   # between-trial precision = (1/between-trial variance)
#
# REGRESSION FOR RECOVERY
for (j in 1:nR){              # LOOP THROUGH STUDIES REPORTING
RECOVERY
   lambda[j,1] <- 0           # treatment effect is zero for control arm
   mu[j] ~ dnorm(0,.0001)      # vague priors for all trial baselines
   for (k in 1:naR[j]) {       # LOOP THROUGH ARMS
      r[j,k] ~ dbin(p[j,k],n[j,k]) # binomial likelihood for recovery data
      logit(p[j,k]) <- mu[j] + lambda[j,k]  # model for linear predictor
      rhat[j,k] <- p[j,k] * n[j,k] # expected value of the numerators
#Deviance contribution
```

```
      devR[j,k] <- 2 * (r[j,k] * (log(r[j,k])-log(rhat[j,k]))
         +  (n[j,k]-r[j,k]) * (log(n[j,k]-r[j,k]) - log(n[j,k]-rhat[j,k])))      }
   for (k in 2:naR[j]) {       # LOOP THROUGH ARMS
# regression equation: LOR(Rec) = beta * LOR(SMD)
      lambda[j,k] <- beta * (-3.1416/sqrt(3)) * delta[C[j],k]
   }
#  summed residual deviance contribution for this trial
   resdevR[j] <- sum(devR[j,1:naR[j]])
}
totresdevR <- sum(resdevR[])     #Total Residual Deviance Recovery
beta ~ dnorm(0,.0001)
# get confidence bound for beta for plotting
for (j in 1:1001){
   X[j] <- -1 + (j-1)*0.005
   Y[j] <- beta * X[j]
 }
#
for (k in 1:nc)  {  mCal[k] <- (-3.1416/sqrt(3)) * beta * m[k]  }
for (k in 1:nt)  {
   g[k] <- (-3.1416/sqrt(3)) * d[k]  # Transform SMD to LOR (change sign)
   gCal[k] <- beta * g[k]   # Transform SMD to LOR (change sign): Calibrated
# rank all treatments on SMD
   rk[k]  <- rank(d[],k)       # Smallest is best (i.e. rank 1)
   best[k]  <- equals(rk[k],1)
   for (h in 1:nt) { prob[h,k] <- equals(rk[k],h) }
# rank all treatments on LOR (calibrated)
   rk.g[k]  <- nt+1-rank(gCal[],k)  # Largest is best (i.e. rank 1)
   best.g[k]  <- equals(rk.g[k],1)
   for (h in 1:nt) { prob.g[h,k] <- equals(rk.g[k],h) }
  }
for (c in 1:(nt-1)) {           # for each treatment
  for (k in (c+1):nt) {
   diff[c,k] <- d[k] - d[c]     # pairwise SMDs
   lor[c,k] <- g[k] - g[c]      # pairwise LORs
   or[c,k] <- exp(g[k] - g[c])  # Paiwise ORs
   lorCal[c,k] <- gCal[k] - gCal[c]     # pairwise LORs (calibrated)
   orCal[c,k] <- exp(gCal[k] - gCal[c]) # Paiwise ORs (calibrated)
   rr[c,k] <- RR[k]/RR[c]
  }
 }
for (c in 1:(nc-1)) {           # for each class
  for (k in (c+1):nc) {
   diffClass[c,k] <- m[k] - m[c]          # class v class SMDs
   lorClass[c,k] <- mCal[k] - mCal[c]     # class v class LORs (calibrated)
   orClass[c,k] <- exp(mCal[k] - mCal[c]) # class v class ORs (calibrated)
   rrClass[c,k] <- RRC[k]/RRC[c]  # RR for class
```

```
  }
 }
# Provide estimates of treatment effects T[k] on the natural scale
# Given a Mean Effect, meanA, for 'standard' treatment A,
# with precision (1/variance) precA
A ~ dnorm(meanA,precA)
for (k in 1:nt) {
   logit(T[k]) <- A + gCal[k]    # prob recovery for each treatment
   RR[k] <- T[k]/T[1]            # relative risk for each treatment
  }
for (k in 1:nc) {
   logit(TClass[k]) <- A + mCal[k] # prob recovery for each class
   RRC[k] <- TClass[k]/TClass[1] # relative risk for each class
 }
}                                # *** PROGRAM ENDS
```