

Appendix 1 WinBUGS code for network meta-analysis

```

# random effects model for multi-arm trials
model{
  for(i in 1:ns){
    w[i,1] <- 0
    delta[i,1] <- 0
    mu[i] ~ dnorm(0,.0001)
    for (k in 1:na[i]) {
      r[i,k] ~ dbin(p[i,k],n[i,k])
      logit(p[i,k]) <- mu[i] + delta[i,k]
      rhat[i,k] <- p[i,k] * n[i,k]
      dev[i,k] <- 2 * (r[i,k] * (log(r[i,k])-log(rhat[i,k]))) +
        (n[i,k]-r[i,k]) * (log(n[i,k]-r[i,k]) - log(n[i,k]-rhat[i,k])))
    }
    resdev[i] <- sum(dev[i,1:na[i]])
  }
  this trial
  for (k in 2:na[i]) {
    delta[i,k] ~ dnorm(md[i,k],taud[i,k])
    md[i,k] <- d[t[i,k]] - d[t[i,1]] + sw[i,k]
    taud[i,k] <- tau *2*(k-1)/k
    w[i,k] <- (delta[i,k] - d[t[i,k]] + d[t[i,1]])
    sw[i,k] <- sum(w[i,1:k-1])/(k-1)
  }
  trials
}
}
totresdev <- sum(resdev[])
d[1] <- 0
treatment
for (k in 2:nt){
  d[k] ~ dnorm(0,.0001)
}
sd ~ dunif(0,5)
tau <- pow(sd,-2)

# pairwise ORs and LORs for all possible pair-wise comparisons
for (c in 1:(nt-1)){
  for (k in (c+1):nt){
    or[c,k] <- exp(d[k] - d[c])
    lor[c,k] <- (d[k]-d[c])
  }
}

# ranking on relative scale
for (k in 1:nt) {
  rk[k] <- nt+1-rank(d[],k)
  best[k] <- equals(rk[k],1)
  for (j in 1:nt) {
    effectiveness[k,j] <- equals(rk[k],j )
  }
}

for (k in 1:nt) {
  for (j in 1:nt) {

```

*** PROGRAM STARTS
 # LOOP THROUGH STUDIES
 # adjustment for multi arm trial
 # treatment effect is zero for control arm
 # vague priors for all trial baselines
 # LOOP THROUGH ARMS
 # binomial likelihood
 # model for linear predictor
 # expected value of the numerators
 # Deviance contribution
 # summed residual deviance contribution for this trial
 # LOOP THROUGH ARMS
 # trial-specific LOR distributions
 # mean of LOR distributions
 #precision of LOR distributions
 #adjustment, multi-arm RCTs
 # cumulative adjustment for multi-arm
 #Total Residual Deviance
 # treatment effect is zero for reference
 # vague priors for treatment effects
 #calculate probability that treat k is best

```
cumeffectiveness[k,j] <- sum(effectiveness[k, 1:j])
}
}

# SUCRAS
for (k in 1:nt) {
  SUCRA[k] <- sum(cumeffectiveness[k,1:(nt-1)])/(nt-1)
}
}

# *** PROGRAM ENDS
```