R script code for the performed latent class analysis:

5P3T_dep_default_with_priors.

```
Model
{
# multinomial models
for (k in 1:K){
  y[k,1:2,1:2,1:2] ~ dmulti(p[k,1:2,1:2,1:2], n[k])
  }

# cell probabilities expressed in terms of se, sp and pi
for (k in 1:K){
  p[k,1,1,1] <- pi[k]*se[1]*(se[2]*se[3]+cov23[1]) + (1-pi[k])*(1-sp[1])*((1-sp[2])*(1-sp[3])+cov23[2])
  p[k,1,1,2] <- pi[k]*se[1]*(se[2]*(1-se[3])-cov23[1]) + (1-pi[k])*(1-sp[1])*((1-sp[2])*sp[3]-cov23[2])
  p[k,1,2,1] <- pi[k]*se[1]*((1-se[2])*se[3]-cov23[1]) + (1-pi[k])*(1-sp[1])*(sp[2]*(1-sp[3])-cov23[2])
  p[k,1,2,2] <- pi[k]*se[1]*((1-se[2])*(1-se[3])+cov23[1]) + (1-pi[k])*(1-sp[1])*(sp[2]*sp[3]+cov23[2])
  p[k,2,1,1] <- pi[k]*(1-se[1])*(se[2]*se[3]+cov23[1]) + (1-pi[k])*sp[1]*((1-sp[2])*(1-sp[3])+cov23[2])
  p[k,2,1,2] <- pi[k]*(1-se[1])*(se[2]*(1-se[3])-cov23[1]) + (1-pi[k])*sp[1]*((1-sp[2])*sp[3]-cov23[2])
  p[k,2,2,1] <- pi[k]*(1-se[1])*((1-se[2])*se[3]-cov23[1]) + (1-pi[k])*sp[1]*(sp[2]*(1-sp[3])-cov23[2])
  p[k,2,2,2] <- pi[k]*(1-se[1])*((1-se[2])*(1-se[3])+cov23[1]) + (1-pi[k])*sp[1]*(sp[2]*sp[3]+cov23[2])
  }

# prior distributions
se[1] ~ dbeta(1,1)
sp[1] ~ dbeta(1,1)
se[2] ~ dbeta(1,1)
sp[2] ~ dbeta(1,1)
se[3] ~ dbeta(30.665,9.367)
sp[3] ~ dbeta(100,1.497)
for (k in 1:K){
  pi[k] ~ dbeta(1,1)
  }
```

```
# Covariance note: cov12[1] ~ sensitivity, cov12[2] ~ specificity
cov23_low[1] <- max(-(1-se[2])*(1-se[3]),-se[2]*se[3])
cov23_upp[1] <- min(se[2]*(1-se[3]),(1-se[2])*se[3])
cov23_low[2] <- max(-(1-sp[2])*(1-sp[3]),-sp[2]*sp[3])
cov23_upp[2] <- min(sp[2]*(1-sp[3]),(1-sp[2])*sp[3])
cov23[1] ~ dunif(cov23_low[1],cov23_upp[1])
cov23[2] ~ dunif(cov23_low[2],cov23_upp[2])

#Step function
sediff[1] <- se[1]- se[2]
sediff[2] <- se[1]- se[3]
sediff[3] <- se[2]- se[1]
sediff[4] <- se[2]- se[3]
sediff[5] <- se[3]- se[1]
sediff[6] <- se[3]- se[2]

spdiff[1] <- sp[1]- sp[2]
spdiff[2] <- sp[1]- sp[3]
spdiff[3] <- sp[2]- sp[1]
spdiff[4] <- sp[2]- sp[3]
spdiff[5] <- sp[3]- sp[1]
spdiff[6] <- sp[3]- sp[2]

Prsediff[1] <- step(sediff[1])
Prsediff[2] <- step(sediff[2])
Prsediff[3] <- step(sediff[3])
Prsediff[4] <- step(sediff[4])
Prsediff[5] <- step(sediff[5])
Prsediff[6] <- step(sediff[6])
```

```
Prspdiff[1] <- step(spdiff[1])

Prspdiff[2] <- step(spdiff[2])

Prspdiff[3] <- step(spdiff[3])

Prspdiff[4] <- step(spdiff[4])

Prspdiff[5] <- step(spdiff[5])

Prspdiff[6] <- step(spdiff[6])



# 3-test model: Bayesian p-value for Pearson X2

for (k in 1:K){

 s[k,1:2,1:2,1:2] ~ dmulti(p[k,1:2,1:2,1:2],n[k])

 for (i1 in 1:2){

  for (i2 in 1:2){

   for (i3 in 1:2){

    x2[k,i1,i2,i3] <- pow(y[k,i1,i2,i3]-p[k,i1,i2,i3]*n[k],2)/(p[k,i1,i2,i3]*n[k])

    ref[k,i1,i2,i3] <- pow(s[k,i1,i2,i3]-p[k,i1,i2,i3]*n[k],2)/(p[k,i1,i2,i3]*n[k])

 }}}}

x2total <- sum(x2[,,,])

reftotal <- sum(ref[,,,])

bayesp <- step(reftotal-x2total)



}



# note: required order of cell counts:

#    (test1=+,test2=+,test3=+), (+,+,-), (+,-,+), (+,-,-), (-,+,+), (-,+,-), (-,-,+), (-,-,-)

#    for popul 1,2,...

# note: first number in c() list for .Dim is the number of populations (K)

list(K=5, n=c(93,178,180,92,51),
y=structure(.Data=c(0,0,0,0,0,5,0,88,2,0,0,1,0,1,0,173,0,0,0,4,3,10,1,162,1,1,0,7,7,13,0,63,0,0,0,0,0,0,0,
51),.Dim=c(5,2,2,2)))
```