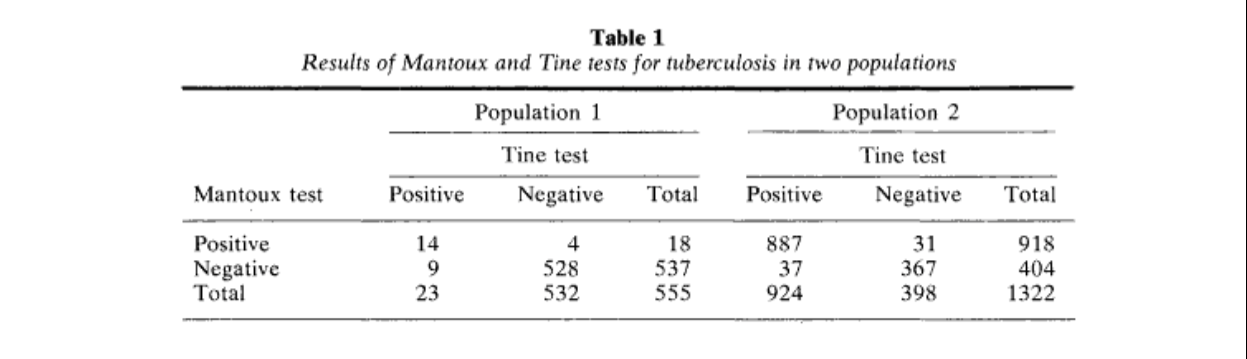
**Exercise 2 – The Hui-Walter model**

In this session we will apply a Bayesian Latent class model to the data/observations from the manuscript published back in 1980 (Table 1).



Estimating the Error Rates of Diagnostic Tests, S. L. Hui and S. D. Walter, 1980

1. Add population 1 of table 1 in your working directory, with the following lines of code.

pop\_1 = matrix(nrow=3,ncol=3)

rownames(pop\_1) = c("Mantoux\_Test\_Pos", "Mantoux\_Test\_Neg", "Total")

colnames(pop\_1) = c("Tine\_Test\_Pos", "Tine\_Test\_Neg", "Total")

pop\_1[1,1] = 14

pop\_1[1,2] = 4

pop\_1[2,1] = 9

pop\_1[2,2] = 528

#Total rows and columns

pop\_1[1,3] = pop\_1[1,1] + pop\_1[1,2]

pop\_1[2,3] = pop\_1[2,1] + pop\_1[2,2]

pop\_1[3,1] = pop\_1[1,1] + pop\_1[2,1]

pop\_1[3,2] = pop\_1[1,2] + pop\_1[2,2]

N\_1 = sum(pop\_1[1,1] + pop\_1[1,2] + pop\_1[2,1] + pop\_1[2,2])

pop\_1[3,3] = N\_1

pop\_1

1. Do the same with pop\_2

pop\_2 = matrix(nrow=3,ncol=3)

rownames(pop\_2) = c("Mantoux\_Test\_Pos", "Mantoux\_Test\_Neg", "Total")

colnames(pop\_2) = c("Tine\_Test\_Pos", "Tine\_Test\_Neg", "Total")

pop\_2[1,1] = 887

pop\_2[1,2] = 31

pop\_2[2,1] = 37

pop\_2[2,2] = 367

#Total rows and columns

pop\_2[1,3] = pop\_2[1,1] + pop\_2[1,2]

pop\_2[2,3] = pop\_2[2,1] + pop\_2[2,2]

pop\_2[3,1] = pop\_2[1,1] + pop\_2[2,1]

pop\_2[3,2] = pop\_2[1,2] + pop\_2[2,2]

N\_2 = sum(pop\_2[1,1] + pop\_2[1,2] + pop\_2[2,1] + pop\_2[2,2])

pop\_2[3,3] = N\_2

pop\_2

1. Write the 2\_test\_2\_population model and save as “hw\_blca.bug”.

model{

Population\_1 ~ dmulti(prob\_1, N\_1)

#Population\_1

# Test1+ Test2+

prob\_1[1] <- (prev[1] \* ((se[1])\*(se[2]))) + ((1-prev[1]) \* ((1-sp[1])\*(1-sp[2])))

# Test1+ Test2-

prob\_1[2] <- (prev[1] \* ((se[1])\*(1-se[2]))) + ((1-prev[1]) \* ((1-sp[1])\*(sp[2])))

# Test1- Test2+

prob\_1[3] <- (prev[1] \* ((1-se[1])\*(se[2]))) + ((1-prev[1]) \* ((sp[1])\*(1-sp[2])))

# Test1- Test2-

prob\_1[4] <- (prev[1] \* ((1-se[1])\*(1-se[2]))) + ((1-prev[1]) \* ((sp[1])\*(sp[2])))

#Population\_2

Population\_2 ~ dmulti(prob\_2, N\_2)

# Test1+ Test2+

prob\_2[1] <- (prev[2] \* ((se[1])\*(se[2]))) + ((1-prev[2]) \* ((1-sp[1])\*(1-sp[2])))

# Test1+ Test2-

prob\_2[2] <- (prev[2] \* ((se[1])\*(1-se[2]))) + ((1-prev[2]) \* ((1-sp[1])\*(sp[2])))

# Test1- Test2+

prob\_2[3] <- (prev[2] \* ((1-se[1])\*(se[2]))) + ((1-prev[2]) \* ((sp[1])\*(1-sp[2])))

# Test1- Test2-

prob\_2[4] <- (prev[2] \* ((1-se[1])\*(1-se[2]))) + ((1-prev[2]) \* ((sp[1])\*(sp[2])))

prev[1] ~ dbeta(1, 1)

prev[2] ~ dbeta(1, 1)

se[1] ~ dbeta(1, 1)T(1-sp[1], )

sp[1] ~ dbeta(1, 1)

se[2] ~ dbeta(1, 1)T(1-sp[2], )

sp[2] ~ dbeta(1, 1)

#data# Population\_1, Population\_2, N\_1, N\_2

#monitor# prev, prob\_1, prob\_2, se, sp

#inits# prev, se, sp

}

1. Attach the “runjags” package

library(runjags)

1. Specify the data in R environment

Population\_1 <- as.numeric(pop\_1[1:2,1:2])

Population\_2 <- as.numeric(pop\_2[1:2,1:2])

1. Define the initial values data for the MCMC chains:

prev <- list(chain1=c(0.05,0.99), chain2=c(0.95,0.05))

se <- list(chain1=c(0.5,0.99), chain2=c(0.99,0.5))

sp <- list(chain1=c(0.5,0.99), chain2=c(0.99,0.5))

1. Run the model

results <- run.jags(“hw\_blca.bug”, n.chains=2, burnin=5000, sample=10000)

1. Plot and view the results

summary(results)

pt <- plot(results)

pt$`prev[1].plot1`

pt$`prev[1].plot3`

print(pt[["prev[1].plot1"]])

print(pt[["se[1].plot1"]])

print(pt[["sp[1].plot1"]])

print(pt[["sp[1].plot3"]])