#=======MODEL TO ESTIMATE TRUE BETWEEN-HOUSEHOLD PREVALENCE============

#------------------------- MODEL AIM ---------------------------------

## The aim of the model is to use serosurvey results to estimate the true between-household prevalence and 95% credibility interval, taking into account:

# 1. Uncertainty arising from using an imperfect test

# 2. Uncertainty in the sensitivity and specificity of the test (if data is available)

# 3. Uncertainty arising from sampling some but not necessarily all of the animals within the household.

#-------------------------INSTRUCTIONS------------------------------

## This script can be run in R: https://www.r-project.org/

## In order to run the code, save a .csv file with one row for each farm that was sampled; the second column should list the numbers of eligible animals in each household, the third column the number of animals in the household that were sampled, and the fourth, the number of animals that tested positive. Save the file as "data.csv"

## Parameters that need to be changed for each study are indicated by "#•"

#-------------------------MODEL OVERVIEW------------------------------

### PARAMETERS ###

# N = the total number of eligible animals in the household

# n = the total number of animals in the household that were sampled

# Test\_P = the total number of animals in the household that tested positive

# T\_Sn = number of "true positive" samples used to estimate sensitivity of the test

# P\_Sn = number of "true positive" samples that were correctly identified as positive by the test

# T\_Sp = number of "true negative" samples used to estimate specificity of the test

# P\_Sp = number "true negative" samples that were correctly identified as negative by the test

### STEP A ###

# For each farm, generate all possible numbers of positive animals on the farm

### STEP B ###

# Calculate the "prior probability" of each of the numbers generated in STEP A

### STEP C ###

# Generate all possible numbers of tested animals that were true positives.

### STEP D ###

# Calculate the likelihoods of getting the test results for each of the numbers generated in STEP C

### STEP E ###

# Calculate the posterior odds that the farm was truly negative.

### STEP F ###

# Simulate whether the farm is positive or negative.

### STEP G ###

# Repeat step F 10000 times to create an uncertainty distribution of true household prevalence in the sample. Use the distribution to estimate the true household prevalence and 95 per cent credible interval.

#======================== MODEL ======================================

### INSTALL OR LOAD PACKAGES ###

library(epiR) # only needed for 95% confidence interval at end

#set working directory

setwd("") #•

#load data

data <- read.csv("data.csv")

### PARAMETERS ###

# N = the total number of eligible animals in the household

# n = the total number of animals in the household that were sampled

# Test\_P = the total number of animals in the household that tested positive

# T\_Sn = number of "true positive" samples used to estimate sensitivity of the test

# P\_Sn = number of "true positive" samples that correctly tested positive

# T\_Sp = number of "true negative" samples used to estimate specificity of the test

# P\_Sp = number "true negative" samples that correctly tested negative

# No\_households = the number of households that were sampled

# No\_simulations = the number of simulations

No\_households <- 22 #•#

No\_simulations <- 10000 #•#

T\_Sn <- 18 #•#

P\_Sn <- 18 #•#

T\_Sp <- 650 #•#

P\_Sp <- 650 #•#

N <- data[c(1:No\_households),2]

n <- data[c(1:No\_households),3]

Test\_P <- data[c(1:No\_households),4]

# Calculated parameters

## N\_HH = Number of households

N\_HH <- length(N)

## prior distirbution of within-farm prevalence values

individual\_prevalences <- numeric(0)

for (i in 1:N\_HH) {

individual\_prevalence <- Test\_P[i]/n[i]

individual\_prevalences <- c(individual\_prevalences, individual\_prevalence)

}

## create vector for simulated household prevalence values

HH\_prevalence\_values <- numeric(0)

for (simulations in 1:No\_simulations) {

## create vector for household states (positive or negative)

HH\_states <- numeric(0)

# calculated parameters

## Estimate sensitivity (Sn) and specificity (Sp) from experimental data [or absolute values could be assigned here]

Sn <- rbeta(1,P\_Sn+1,T\_Sn-P\_Sn+1)

Sp <- rbeta(1,P\_Sp+1,T\_Sp-P\_Sp+1)

## FPF = The false positive fraction i.e. (1-Sp)

FPF <- 1-Sp

for (HH\_i in 1:N\_HH) {

# Create a vector for the values of prior probability household is negative \* likelihood household is negative

prior\_by\_likelihood\_neg <- numeric(0)

# Create a vector for the values of prior probability household is positive \* likelihood household is negative

prior\_by\_likelihood\_pos <- numeric(0)

### STEP A ###

# For each household, generate all possible numbers of true positive animals on the household (TPH)

TPH <- c(0:N[HH\_i])

# Work out the corresponding number of true negative animals on the household (TNH)

TNH <- N[HH\_i] - TPH

### STEP B ###

# Calculate the prior "probability" of each of the numbers generated in STEP A (in other words, the relative frequency of that particular within household prevalence)

# Round individual prevalence values (individual\_prevalences) to the equivalent number of positives on this farm

adjusted\_individual\_prevalences <- round(individual\_prevalences\*N[HH\_i])

priors <- numeric(0)

min\_TPS <- numeric(0)

max\_TPS <- numeric(0)

TPS <- list(0)

TNS <- list(0)

for(TPH\_i in 1:length(TPH)){

prior <- sum(adjusted\_individual\_prevalences == TPH[TPH\_i])/length(adjusted\_individual\_prevalences)

### STEP C ###

# For each number of TPH, generate all possible numbers of tested animals that could be true positives (TPS).

if(TNH[TPH\_i] >= n[HH\_i]) {

min\_TPS <- 0

} else {

min\_TPS <- n[HH\_i] - TNH[TPH\_i]

}

if(n[HH\_i] <= TPH[TPH\_i]) {

max\_TPS <- n[HH\_i]

} else{

max\_TPS <- TPH[TPH\_i]

}

TPS <- as.numeric(c(min\_TPS:max\_TPS))

# Generate the corresponding numbers of true negative (TNS)

TNS <- n[HH\_i] - TPS

### STEP D ###

# For each number of true positive sampled animals (TPS) Generate all possible numbers of true test positive animals

min\_TPS\_Test\_P <- numeric(0)

max\_TPS\_Test\_P <- numeric(0)

TPS\_Test\_P <- list(0)

for (i in 1:length(TPS)) {

## calculate the maximum possible value of TPS\_Test\_P

if(Test\_P[HH\_i] > TPS[i]){

max\_TPS\_Test\_P <- TPS[i]

} else {

max\_TPS\_Test\_P <- Test\_P[HH\_i]

}

## calculate the minimum possible value of TPS\_Test\_P

if(Test\_P[HH\_i] < TNS[i]){

min\_TPS\_Test\_P <- 0

} else {

min\_TPS\_Test\_P <- Test\_P[HH\_i] - TNS[i]

}

##create a vector for values of TPS\_Test\_P

TPS\_Test\_P <- c(min\_TPS\_Test\_P:max\_TPS\_Test\_P)

# For each number of true positive sampled animals (TPS) Generate all possible numbers of false test positive animals (False\_Test\_P)

False\_Test\_P <- Test\_P[HH\_i] - TPS\_Test\_P

# Calculate the likelihoods of getting the results for each of the numbers generated in STEP D

for (i2 in 1:length(TPS\_Test\_P)){

likelihood <- dhyper(TPS[i],TPH[TPH\_i],TNH[TPH\_i],n[HH\_i])\*dbinom(TPS\_Test\_P[i2],TPS[i],Sn)\*dbinom(False\_Test\_P[i2],TNS[i],FPF)

likelihood\_by\_prior\_prob <- likelihood\*prior

if (TPH[TPH\_i] >0) {

prior\_by\_likelihood\_pos <- c(prior\_by\_likelihood\_pos,likelihood\_by\_prior\_prob)

} else {

prior\_by\_likelihood\_neg <- c(prior\_by\_likelihood\_neg,likelihood\_by\_prior\_prob)

}

}

}

}

### STEP E ###

# Calculate the posterior odds that the farm was truly negative, and then the probability.

posterior\_odds <- sum(prior\_by\_likelihood\_neg)/sum(prior\_by\_likelihood\_pos)

probability\_farm\_neg <- posterior\_odds/(1 + posterior\_odds)

### STEP F ###

# Simulate whether the farm is positive or negative.

HH\_state <- rbinom(1,1,(1-probability\_farm\_neg))

HH\_states <- c(HH\_states,HH\_state)

}

### STEP G ###

# Repeat step D 10000 times to create an uncertainty distribution of true household prevalence. Use the distribution to estimate the true household prevalence and 95 per cent confidence interval.

HH\_prevalence <- sum(HH\_states)/N\_HH

HH\_prevalence\_values <- c(HH\_prevalence\_values,HH\_prevalence)

}

hist(HH\_prevalence\_values)

epi.descriptives(HH\_prevalence\_values, quantile = c(0.025, 0.975))