Probabilility of detections and sample occurance

Richard A. Erickson, Chris M. Merkes, and Erica L. Mize

Derviation of probabilty calucation

Calcuating the probabilty of eDNA occuring given two levels of sampling is not straight forward. Rather than calculating the probabilty of detecting eDNA, we calculate the probabiltiy of non-detecting DNA and then subtract it from 1. The probability of eDNA occuring is a sample is θ. The probability of detecting eDNA within a sample given DNA is present within the sample is p. The probabitly of not detecting eDNA within a sample give mutiple samples K may be written as $1-(1-p)^K$. This calculation for non-detecting eDNA is broken down into two parts. First, the probabtily of not detecting eDNA because it truely is not in the sample needs to be calculated, which is $1-\theta$, for a given sample. Second, the probabitliy of missing eDNA even though the eDNA is present within the sample needs to be calculated as well: $\theta(1-p)^K$.

For the case where only 1 sample is take (i.e., $J=1$), the probability of not detecting eDNA in any sample of subsample may be written:

$P(y_{j,k}=0|\theta,p,k)=1-\theta+\theta(1-p)^k$.

For the case when 2 samples are taken (i.e, $J=2$), the probability of not detecting eDNA in any of the subsampels may be written as:

$P(y_{j,k}=0|\theta,p,k)=(1-\theta)^2+2(1-\theta)(\theta(1-p)^k)+(\theta(1-p)^k)^2$.

For $J=3$, it follows that:

$P(y_{j,k}=0|\theta,p,k)=(1-\theta)^3+3(1-\theta)^2(\theta(1-p)^k)+3(1-\theta)(\theta(1-p)^k)^2+(\theta(1-p)^k)^3$.

In turn, this generalizes to be

$P(y_{j,k}=0|\theta,p,k)=\sum_{j=1}^{J}=\binom{J}{j}(1-\theta)^j(\theta(1-p)^k)^{J-j}$.

Data source and parameter values

The observation and detection probabilities are based upon "rough" analysis from USGS eDNA samples in Pools 17 and 19. These samples were taken during June, which should be a period with low detentions based upon the biology and seasonal movement of the fish. The lowest observed occurrence probabilities were 0.06 and the highest was 0.42. These values are used for the basis of a sample containing DNA. The probabilities of detecting DNA within a sample ranged from 0.3 to 0.4. These values are used for the basis of detecting DNA using an assay for a given sample.

Probability of detecting a species (Occupuancy only)

The first analysis we run estimates the probability of detecting a species at site. This does not allow us to distinguish different densities. Rather it simply informs if a species is present at a site.

We first write a function that estimates the probability of detecting a species assuming different numbers of samples, `J`; probabilities of samples containing DNA, `theta`; different numbers of assay replicates, `K`; and different detection probabilities for the assay, `pDetection` (we choose to use `pDetection` rather than `p` to have a variable that was easier to find in our code). We derived this relationship in a previous section of the document. We aslo define two helper functions, `E(p,k)`, and `combo`.

```
comb = function(n, r){ factorial(n)/(factorial(r) * factorial(n -r ))}

sampleDetectionOne <- function(
```

```
    J = 50,
    K = 8,
    theta = 0.06,
    pDetection  = 0.3

    ){

    jIndx = J:0
    prob =  sum(comb( J, jIndx) *  (1 - theta) ^ jIndx * (theta * (1 - pDetection)^K )  ^ rev(jIndx))
    return(1 - prob)
    }
```

Next, we explore different sample numbers, $J \in 1, 2, \ldots 100$; different assay detection probabilities, $\theta \in \{0.06, 0.24, 0.42, 0.75\}$; different sample detection probabilities, $p \in \{0.3, 0.35, 0.4, 0.75\}$; and different numbers of molecular replicates $K \in \{2, 3, 4, 8, 16\}$.

We use the `data.table` package for storing and manipulating my data.

```
library(data.table)
results <- data.table(expand.grid(J = 1:100,
                           theta = c(0.06, 0.24, 0.42, 0.75), pDetection = c(0.3, 0.35, 0.4, 0.75),
                           K = c(2, 3, 4, 8, 16)))

for(index in 1:nrow(results)){
  results[ index, ProbDetect :=
      sampleDetectionOne(J = J, K = K, theta = theta, pDetection = pDetection)]
}
results[ , thetaPlot := factor(paste0("theta = ", theta))]
results[ , pDetectionPlot := factor(paste0("p = ", pDetection))]
results[ , KPlot := factor(paste0("K = ", K))]
```

Last, we plot the results using `ggplot2`.

```
library(ggplot2)
results[ , KPlot := factor(KPlot,
                           levels = levels(results$KPlot)[order(as.numeric(gsub("K = ", "", levels(resu

detectOne <- ggplot(data = results, aes(x = J, y = ProbDetect, color = KPlot)) +
    geom_line() +
    facet_grid( pDetectionPlot ~ thetaPlot) +
    theme_minimal() +
    ylab("Probabiltiy of detecting species at site") +
    xlab("Number of samples per site") +
  scale_color_manual("Molecular\nreplicates",
                     values = c("red", "blue", "black", "seagreen", "orange"))
print(detectOne)
```

```
ggsave(filename = "detectingOne.pdf", detectOne, width = 6, height = 4)
ggsave(filename = "detectingOne.jpg", detectOne, width = 6, height = 4)
```

Probability of having different observable sample occurancies

A more interesting question than simply detecting species at a site using eDNA is "Can eDNA detect different levels of sample occurrence at sites?". To do this, we conduct a simulation study.

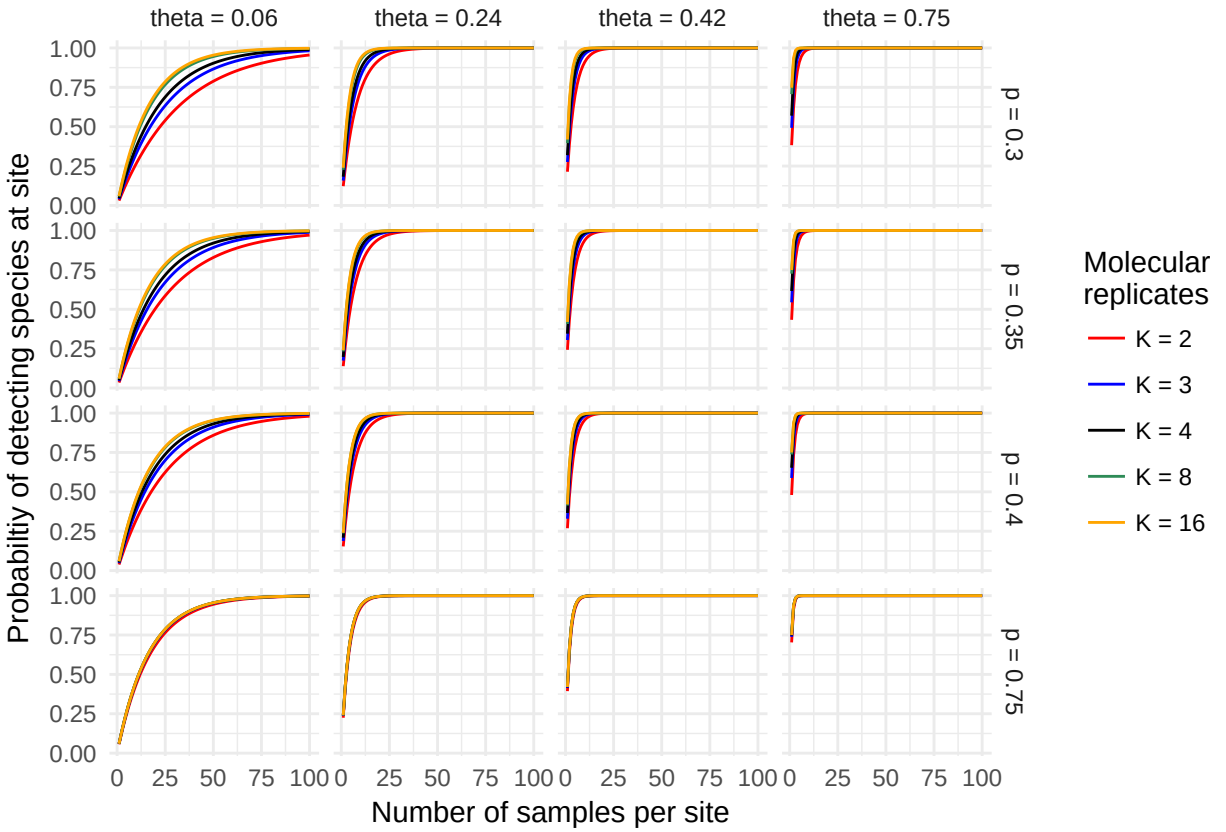

Figure 1: Probability of detecting a species in at least one sample at a site given different assay and sample detection probabilities.

First, we draw samples J from a site with the probability θ of any sample containing DNA from a Bernoulli distribution Bernoulli(J, θ) (Note that is a special case of the binomial distribution with size = 1.). Next, we re-sample the positive samples with K assay replicates with the probability p that an assay detects DNA from a Binomial distribution Binomail(K, p).

```
samplesDetect <- function(
    nSims = 2,
    J = c(10, 100),
    theta = c(0.06, 0.42),
    K = 8,
    pDetection = c(0.3, 0.4)){
        results <- data.table(expand.grid(simulation = 1:nSims,
                        J = J, theta = theta,
                        K = K, pDetection = pDetection))
    for(index in 1:dim(results)[1]){
        results[ index, nPositive :=
            length(which(
                rbinom(n = length(which(rbinom( n = J, size = 1, prob = theta) >0)),
                        size = K, prob = pDetection) > 0))]
        }
    results[ , pPositive := nPositive/J]
    results[ , Samples := factor(paste0("J = ", J))]
    results[ , SamplesPlot  := factor(J)]

    results[ , thetaPlot := factor(paste0("theta = ", theta))]
  results[ , thetaPlot2 := factor( theta)]

  results[ , pDetectionPlot := factor(paste0("p = ", pDetection))]

    factorOrder <- order(as.numeric(gsub("J = ", "", levels(results$Samples))), decreasing = FALSE)
  results[ , Samples := factor(Samples, levels = levels(Samples)[factorOrder]) ]

  results[ , KPlot := paste0("K = ", K)]
  KOrder <- unique(results$KPlot)[order(as.numeric(gsub("K = ", "", unique(results$KPlot))), decreasing
  results[ , KPlot := factor( KPlot, levels = KOrder)]

  results[ , KPlot2 := factor(gsub("K = ", "", KPlot))]
  KOrder2 <- unique(results$KPlot2)[order(as.numeric(unique(results$KPlot)), decreasing = FALSE)]
  results[ , KPlot2 := factor( KPlot2, levels = KOrder2)]

    return(results)
    }
```

Next, we explore different sample numbers, $n \in \{20, 50, 100\}$; different assay detection probabilities, $p \in \{0.15, 0.3, 0.35, 0.4, 0.75\}$; and different sample detection probabilities, $\theta \in \{0.06, 0.24, 0.42, 0.76\}$ by running 4,000 simulations.

```
sampleResults <- samplesDetect(nSims = 4000,
                               theta = c(0.06, 0.24, 0.42, 0.76),
                               pDetection = c(0.15, 0.3, 0.35, 0.4, 0.75),
                               J = c(5, 20, 50, 100),
                               K = c(2, 3, 4, 8, 16))
```

Last, we plot the results using `ggplot2`

```
compareSites <- ggplot(sampleResults, aes(x = KPlot2, y = pPositive, fill = thetaPlot)) +
    geom_boxplot(outlier.size = 0.5) +
    facet_grid(  Samples ~  pDetectionPlot ) +
  theme_minimal() +
    ylab(expression(over("Number of simulated positive samples","Total number of simulated samples"))) +
  xlab("Number of molecular replicates") +
  scale_fill_manual(expression("Generating "*theta),
                    values = c("red", "blue", "black", "seagreen", "orange"))
print(compareSites)
```

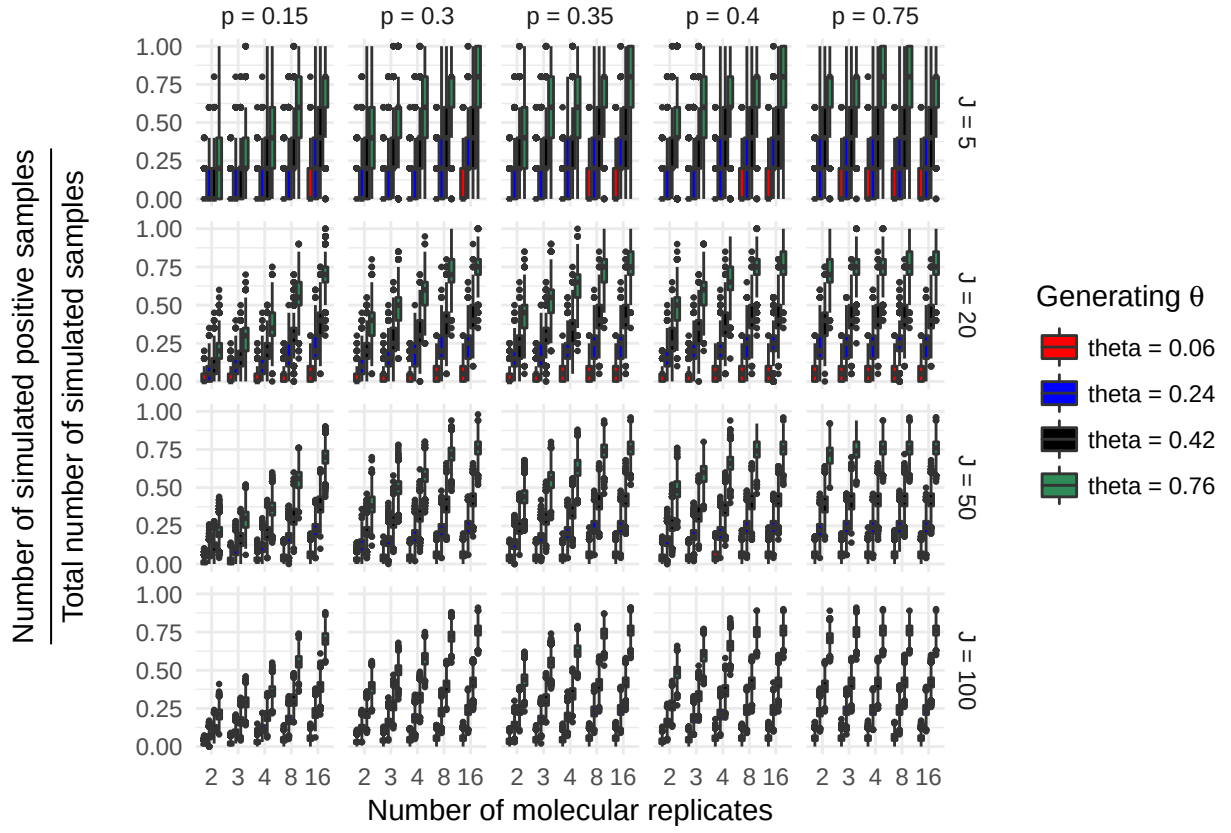

Figure 2: Proportion of samples per site (Sample occurance) that are positive based upon sample size and the assay's probability of detection.

```
ggsave(filename = "compareSites.pdf", compareSites, width = 11, height = 6)
ggsave(filename = "compareSites.jpg", compareSites, width = 11, height = 6)
```