

Probabilility of detections and sample occurance

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

Metadata note

See `sampleSize.Rmd` for meta-data. This file contains script to generate plots used for posters and presentations on the results from this manuscript.

This file is indented to allow the recreation of figures and assumes the reader can understand R code with minimal documentation.

```
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)
}

library(data.table)
results <- data.table(expand.grid(J = 1:100,
  theta = c(0.05, 0.1, 0.2, 0.4, 0.8, 1.0),
  pDetection = c(0.05, 0.1, 0.2, 0.4, 0.8, 1.0),
  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))]

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

detectOne <- ggplot(data = results[ theta == 0.05 & pDetection == 0.2,],
  aes(x = J, y = ProbDetect, color = KPlot)) +
  geom_line() +
  theme_minimal() +
  ylab("Probabilitiy of detecting species at site") +
  xlab("Number of samples per site") +
  scale_color_manual("Molecular\nreplicates",
```

```

values = c("red", "blue", "black", "seagreen", "orange", "grey50"))
print(detectOne)

```

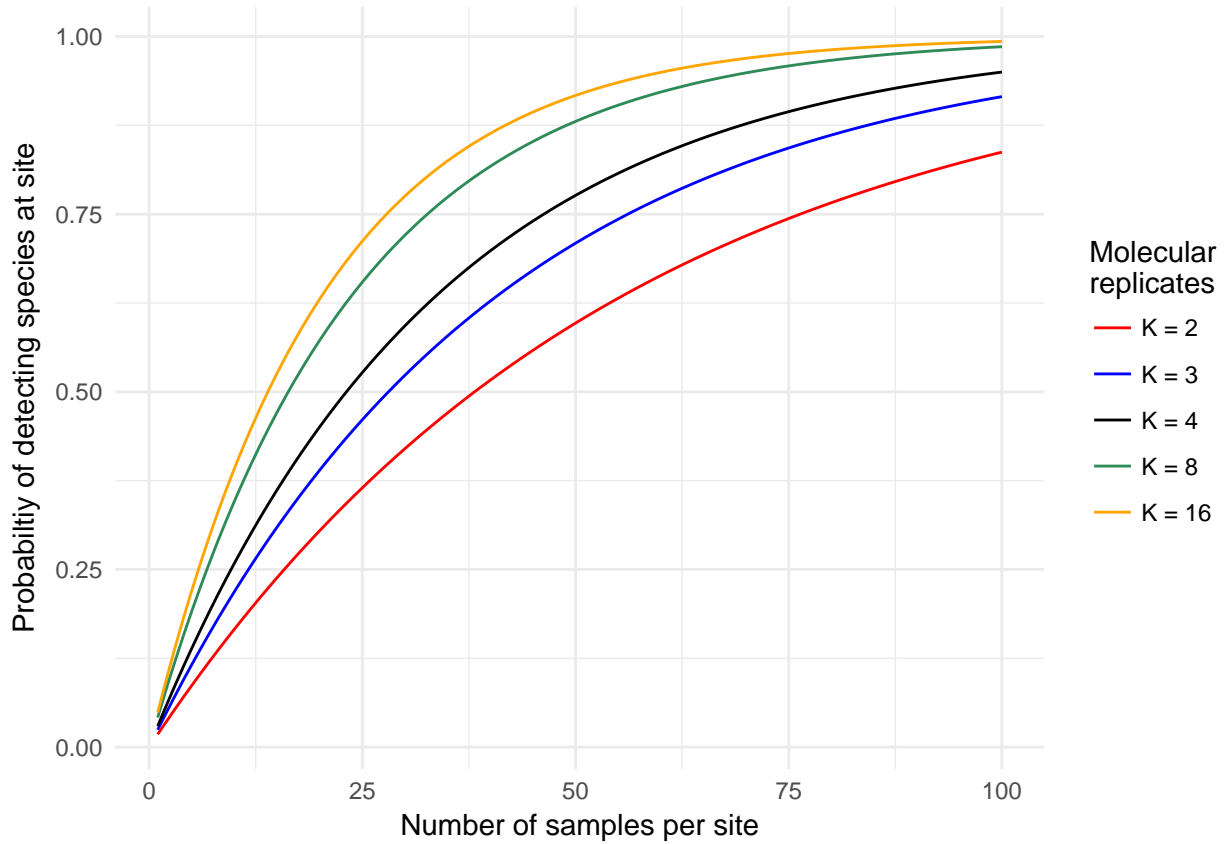


Figure 1: Probability of detecting a species in at least one sample at a site given different assay and sample detection probabilities.

```

ggsave(filename = "detectingOneP3T06.pdf", detectOne, width = 6, height = 4)
ggsave(filename = "detectingOneP3T06.jpg", detectOne, width = 6, height = 4)

```

```

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[, 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 = FALSE)]
    results[ , KPlot := factor( KPlot, levels = KOrder)]

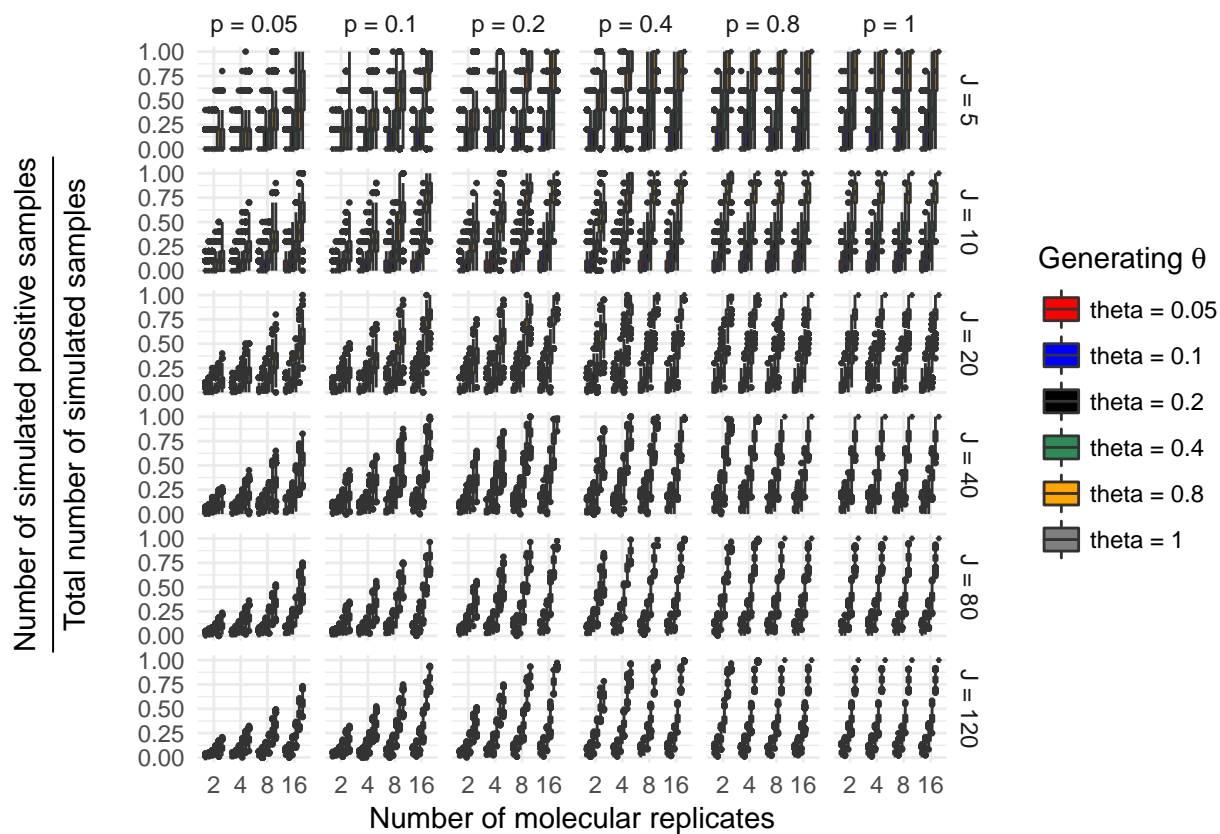
    results[ , KPlot2 := factor(gsub("K = ", "", KPlot))]
    KOrder2 <- unique(results$KPlot2)[order(as.numeric(unique(results$KPlot)), decreasing = FALSE)]
    results[ , KPlot2 := factor( KPlot2, levels = KOrder2)]

    return(results)
}

sampleResults <- samplesDetect(nSims = 4000,
                               theta = c(0.05, 0.1, 0.2, 0.4, 0.8, 1.0),
                               pDetection = c(0.05, 0.1, 0.2, 0.4, 0.8, 1.0),
                               J = c(5, 10, 20, 40, 80, 120),
                               K = c(2, 4, 8, 16))

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", "grey50"))
print(compareSites)

```



```
ggsave(filename = "compareSitesPoster.pdf", compareSites, width = 6, height = 4)
ggsave(filename = "compareSitesPoster.jpg", compareSites, width = 6, height = 4)
```