North American Journal of Fisheries Management 37:970–980, 2017

DOI: https://doi.org/10.1080/02755947.2017.1342723

Supplement C: Example R Code for Conducting Sample Size Estimates for the Number of Days to Sample when Estimating Catch using the Multi-Day Estimator for Priest Lake, Idaho

```
#Import data
data<-read.table(data)

#View first few rows of data
head(data)
```

#Number	Date	EffortEst	ObservedCatch	ObservedHoursFished	DayType	Fishery
#1	3/2/2014	0	0	0	WE	Priest
#2	3/6/2014	27	1	3	WD	Priest
#3	3/7/2014	54.13333	8	17	WD	Priest
#4	3/8/2014	285.25	19	60	WE	Priest

```
#Set the number of weekend, week days, and total days in the population
N.we<-104
N.wd<-260
N<-N.we+N.wd

#Define actual catch
catch<-45008

#Calculate season catch rate
CR<-sum(data$ObservedCatch)/sum(data$ObservedHoursFished)

#Subset weekend and weekday data
we.data<-subset(data, DayType=='WE')
wd.data<-subset(data, DayType=='WD')

#Set sample sizes to evaluate
samp.size<-c(seq(2,10,2), seq(10,300,20), seq(310, 1000, 50))

#Set the number of iterations
reps<-5000
#Create an empty vector to store standard error estimates
se.ss<-c()

#Create a for loop to loop over the sample size vector
for(i in 1:length(samp.size)){
#Allocate sample size to weekends and weekdays
n.wd<-round(((5/7)*samp.size[i]))
```

```
n.we<-round(((2/7)*samp.size[i]))
#Create empty vector to store catch rate estimates
catch.est<-c()
        #Create a for loop to loop over iterations
        for(j in 1:reps){
        #Select days to sample with replacement, and conduct the
        #sampling and estimation process
        days.wd<-sample(c(1:nrow(wd.data)), n.wd, T)
        days.we<-sample(c(1:nrow(we.data)), n.we, T)
        wd.effort<-mean(wd.data$EffortEst[days.wd])*N.wd
        we.effort<-mean(we.data$EffortEst[days.we])*N.we
        wd.cr<-
sum(wd.data$ObservedCatch[days.wd])/sum(wd.data$ObservedHoursFished[days.wd])
        we.cr<-
sum(we.data$ObservedCatch[days.we])/sum(we.data$ObservedHoursFished[days.we])
        catch.est[j]<-wd.effort*wd.cr+we.effort*we.cr
        }
#Calculate empirical standard error
se.ss[i]<-sd(catch.est)
}
#Calculate relative confidence interval
CI.percent<-100*((1.96*se.ss)/catch)

#plot results
plot(CI.percent, samp.size, xlim=c(0,60), xlab='95% Relative CI (%)', ylab='Sample size (days)',
type='l')
```