Building Stochastic Search Models for True Proportion, Year, and County Offsets

Hewan Yoseph

2023-20-05

```
data.set <- read.csv("california_hospitalreports2016-2020.csv",header
= TRUE, sep = ",")

pop <- read.csv("ca_pop_2018.csv",header = TRUE,sep=",")


year <- data.set[,1]

county <- data.set[,2]

hospital_name <- data.set[,3]

OSHPDID <- data.set[,4]

hospital_system <- data.set[,5]

type_report <- data.set[,6]

performance <- data.set[,7]

adverse_events <- data.set[,8]

total_cases <- data.set[,9]

longitude <- data.set[,12]

latitude <- as.numeric(as.character(data.set[,13]))


T1 <- length(year)

bare <- matrix(0,nrow=T1,ncol=7)


#populating subsetted data sets
```

```
for (i in 1:T1){

  bare[i,] <-
c(year[i],county[i],hospital_system[i],hospital_name[i],performance[i]
,

                total_cases[i],adverse_events[i])

}


conditions <- unique(bare[,5])

county.names <- unique(bare[,2])

time <- unique(bare[,1])


## lets take out the conditions that do not have significant
information


conditions<- conditions[-6] #Carotid Endarterectomy

conditions <- conditions[-9] #PCI

conditions <- conditions[-(11:18)] #AAA Repair Endo-Unrupture -
Isolated CABG Operative Mortality




## lets take out the counties that do not have significant information


county.names <- county.names[-5] #Colusa

county.names <- county.names[-9] #Glenn

county.names <- county.names[-(14:15)] #Lake and Lassen

county.names <- county.names[-17] # Mariposa
```

```
county.names <- county.names[-(19:20)] #Mono and Modoc

county.names <- county.names[-24] #Plumas

county.names <- county.names[-42] #Trinity



## Also take these counties out of the population data set

pop <- pop[-5,]

pop <- pop[-9,]

pop <- pop[-(14:15),]

pop <- pop[-17,]

pop <- pop[-(19:20),]

pop <- pop[-24,]

pop <- pop[-42,]



K <- length(conditions)

C <- length(county.names)

T <- length(time)


out <- array(0,dim=c(C,K,T,3),dimnames =
list(county.names,conditions,time))


for(t in 1:T)

{

  for(c in 1:C)
```

```
  {
    for(k in 1:K)
    {
      these <-
which((bare[,5]==conditions[k])&(bare[,2]==county.names[c])
                     &(bare[,1]==time[t]))
      out[c,k,t,1] <- sum(strtoi(bare[these,6]),na.rm=TRUE)/pop[c,2]
      out[c,k,t,2] <- sum(strtoi(bare[these,7]),na.rm=TRUE)/pop[c,2]
      out[c,k,t,3] <- (out[c,k,t,2]/out[c,k,t,1])
    }

  }
}

out6 <- out
for(t in 1:T){
  for(c in 1:C){
    for(k in 1:K){

      if(is.na(out6[c,k,t,3])){
        out6[c,k,t,3] <- 0.00
      }
    }
  }
}
```

```
calc.prop.llk <-
function(data,N.events,est.prop,year.offset,county.offset){

  propp <- est.prop+year.offset+county.offset


  llk <- sum(dbinom(data,N.events,propp,log=TRUE))

  return(llk)

}



#this mle will give us the true proportion


L <- 20000

true.prop.results <- NULL


for(k in 1:K){

  res <- matrix(0,nrow=1,ncol=2)

  out5 <- matrix(0,nrow=L,ncol=2)


  N.events <- (out6[,k,,1]) * (pop[,2]) ##total number of cases across
time and space


  d2<- ((out6[,k,,2])*pop[,2])  ##total adverse cases across time and
space
```

```
  out5[1,] <- c(0.0025,-Inf)


  for(l in 2:L){
    u <- runif(1)
    new.est.prop <- out5[l-1,1]


    if(u < 0.5){
      new.est.prop <- new.est.prop + runif(1,-0.0025,0.05)
      }
    else{
      new.est.prop <- new.est.prop
    }


##calculates and stores new likelihood based on new values
    new <- calc.prop.llk(d2,N.events,new.est.prop,0,0)
    if (new > out5[l-1,2]-1){
      out5[l,] <- c(new.est.prop,new)
      }
    else{
      out5[l,] <- out5[l-1]
    }
  }


#MLE indexes
```

```
  maximum.indexes <- which(out5==max(out5[,2]),arr.ind=TRUE)


  true.prop.mle <- out5[maximum.indexes[1],1]

  res <- c(conditions[k],true.prop.mle)


  true.prop.results <- rbind(true.prop.results,res)

  true.prop.results <- as.data.frame(true.prop.results)

}


print(true.prop.results)

########################################################


#true prop is a constant here, only playing around with the offset
values


mle.offset.results <- NULL


for(k in 1:K){

  true <- as.numeric(true.prop.results[k,2])


  for(c in 1:C){


    for(t in 1:T){

      res2 <- matrix(0,nrow=1, ncol=7)
```

```
      outerr <- matrix(0,nrow=L,ncol=4)


      N.events.t <- (out6[c,k,t,1]) * (pop[c,2]) ##total number of
cases in that year across all counties

      d.t<- ((out6[c,k,t,2])*pop[c,2])  ##total adverse cases in that
year across all counties


      outerr[1,] <- c(true,0.0002,0.0002,-10000)


      for(l in 2:L){

        new.year.offset <- outerr[l-1,2]

        new.county.offset <- outerr[l-1,3]


        ut <- runif(1)


        if(ut < 1/2){


          new.year.offset <- new.year.offset + runif(1,-0.0001,0.0001)

          }

        else{


          new.county.offset <- new.county.offset +
runif(1,-0.0001,0.0001)

          }

        newer <-
calc.prop.llk(d.t,N.events.t,true,new.year.offset,new.county.offset)
```

```
        if(newer > (outerr[l-1,4]-1)){

          outerr[l,] <-
c(true,new.year.offset,new.county.offset,newer)
          }

        else{
          outerr[l, ] <- outerr[l-1,]
        }
        }

      maximum.indexes <- which(outerr==max(outerr[,4]),arr.ind=TRUE)

      year.offset.mle <- outerr[maximum.indexes[1],2]
      county.offset.mle <- outerr[maximum.indexes[1],3]
      LL <- outerr[maximum.indexes[1],4]

      res2 <-
c(conditions[k],county.names[c],time[t],true,year.offset.mle,county.of
fset.mle,LL)

      mle.offset.results <- rbind(mle.offset.results,res2)
    }
  }
}
```

```
mle.offset.results <- as.data.frame(mle.offset.results)
```