Multinomial Predictive Belief Functions

Thierry Denoeux

2023-08-22

Preparation

We need the following packages:

```
library(DescTools)
library(evclust)
library(ibelief)
library(lpSolve)
```

Question 1

Let us consider the following probability vector:

```
p <- c(0.3,0.2,0.1,0.4)
```

We generate $q = 1000$ samples from a multinomial distribution with parameters $n = 100$ and p. (This simulates 1000 repetitions of an experiment that consists in drawing 100 balls with replacement from an urn containing balls of different colors, in proportions contained in vector p).

```
set.seed(20230822)
q<-1000
N<-rmultinom(q,100,p)
```

For each of these 1000 samples, we compute the Goodman 95% simultaneous confidence intervals using function `multinomCI` from package `DescTools`, and we count how many times these intervals contain the true probabilities:

```
k<-0
for(i in 1:q){
  int<-MultinomCI(N[,i],conf.level = 0.95)
  if(all((int[,2]<=p)&(int[,3]>=p))) k<-k+1
}
print(k/q)
```

```
## [1] 0.928
```

The coverage rate is close to the nominal value. We can compute a 95% confidence interval on the coverage rate as follows:

```
print(prop.test(k,q)$conf.int)
```

```
## [1] 0.9097395 0.9428833
## attr(,"conf.level")
## [1] 0.95
```

Question 2

We will use function `makeF`from package `evclust`, which gives all the subsets of a frame of K elements, in binary form. For instance, with $K = 3$, we get

```
Foc <- makeF(3,"full")
print(Foc)
```

```
##      [,1] [,2] [,3]
## [1,]    0    0    0
## [2,]    1    0    0
## [3,]    0    1    0
## [4,]    1    1    0
## [5,]    0    0    1
## [6,]    1    0    1
## [7,]    0    1    1
## [8,]    1    1    1
```

In this matrix, each row corresponds to a subset of a frame $\{1, 2, 3\}$ with three elements. The first row corresponds the empty set, the second and third row are, respectively, the singletons $\{1\}$ and $\{2\}$, the fourth row is the pair $\{1, 2\}$, etc. We can remark that there is a symmetry in this table: the last row corresponds to the complement of the first one, row 7 is the complement of row 2, etc.

Let us consider a particular sample, and the corresponding confidence region:

```
K<-3
N<-c(20,30,50)
int<-MultinomCI(N,conf.level=0.95,method="goodman")
```

We can compute the quantities

$$\sum_{\xi_k \in A} P^-(\xi_k) \quad \text{and} \quad \sum_{\xi_k \in A} P^+(\xi_k)$$

for all subsets A as follows:

```
  SPm<-Foc%*%int[,2]
  SPp<-Foc%*%int[,3]
```

The lower and upper probabilities $P^-(A)$ and $P^+(A)$ can then be computed for any subset A as

```
  Pm<-pmax(SPm,1-SPp[2^K:1])
  Pp<-pmin(SPp,1-SPm[2^K:1])
  print(cbind(Pm,Pp))
```

```
##           [,1]      [,2]
## [1,] 0.0000000 0.0000000
## [2,] 0.1217246 0.3107983
## [3,] 0.2036004 0.4180816
## [4,] 0.3835903 0.6164097
## [5,] 0.3835903 0.6164097
## [6,] 0.5819184 0.7963996
## [7,] 0.6892017 0.8782754
## [8,] 1.0000000 1.0000000
```

We know that, for $K = 2$ and $K = 3$, the lower envelope P^- is a belief function, and the upper envelope P^+ is the corresponding plausibility function. We can use function `beltom` in package `ibelief` to compute the corresponding mass function:

```
m<-as.vector(beltom(Pm))
```

Let us print the mass, belief and pausibility functions together with the focal sets:

```
print(cbind(Foc,m,Pm,Pp),2)
```

```
##                m
## [1,] 0 0 0 0.000 0.00 0.00
## [2,] 1 0 0 0.122 0.12 0.31
## [3,] 0 1 0 0.204 0.20 0.42
## [4,] 1 1 0 0.058 0.38 0.62
## [5,] 0 0 1 0.384 0.38 0.62
## [6,] 1 0 1 0.077 0.58 0.80
## [7,] 0 1 1 0.102 0.69 0.88
## [8,] 1 1 1 0.054 1.00 1.00
```

We observe that the masses are all positive, which confirms that P^- is indeed a belief function. Finally, we can write the following generic function that computes m, Bel and Pl for any sample N:

```
multinom_pbf1<-function(N,level=0.95,method_CI="goodman"){
  K<-length(N)
  if(K>3) stop("K must be <=3")
  int<-MultinomCI(N,conf.level=level,method=method_CI)
  Foc<-makeF(K,type="full")
  SPm<-Foc%*%int[,2]
  SPp<-Foc%*%int[,3]
  Pm<-pmax(SPm,1-SPp[2^K:1])
  Pp<-pmin(SPp,1-SPm[2^K:1])
  m<-as.vector(beltom(Pm))
  return(list(m=m,Bel=Pm,Pl=Pp))
}
```

Question 3

Let us now check that the predictive belief function Bel computed from simultaneous confidence intervals is less committed that the probability distribution P_X of X, for at least $100(1-\alpha)\%$ of the samples, i.e.,

$$P(Bel \leq P_X) \geq 1-\alpha.$$

We start with a probability vector p, and compute the corresponding probability measure P_X:

```
p <- c(0.2,0.3,0.5)
P_X <- Foc%*%p
```

We generate 1000 samples from a multinomial random variables N with parameters $n = 100$ and p:

```
q<-1000
N<-rmultinom(q,100,p)
```

We then count how many times Bel is less committed than P_X:

```
k<-0
for(i in 1:q){
  pbf<-multinom_pbf1(as.vector(N[,i]))
  if(all(pbf$Bel<=P_X)) k<-k+1
}
print(k/q)
```

```
## [1] 0.952
```

This is very close to the nominal value 0.95.

Finally, let us check the convergence property,

$$\forall A \subseteq \mathcal{X}, Bel(A) \xrightarrow{P} P_X(A) \text{ as } n \to \infty.$$

For this, we will generate samples with $n = 100$, $n = 1000$ and $n = 10^4$, and plot the distributions of errors $\max_A |Bel(A) - P_X(A)|$:

```
# n=100
N<-rmultinom(q,100,p)
err1<-rep(0,q)
for(i in 1:q){
  pbf<-multinom_pbf1(as.vector(N[,i]))
  err1[i]<-max(abs(P_X-pbf$Bel))
}
# n=1000
N<-rmultinom(q,1000,p)
err2<-rep(0,q)
for(i in 1:q){
  pbf<-multinom_pbf1(as.vector(N[,i]))
  err2[i]<-max(abs(P_X-pbf$Bel))
}
# n=10000
N<-rmultinom(q,10000,p)
err3<-rep(0,q)
for(i in 1:q){
  pbf<-multinom_pbf1(as.vector(N[,i]))
  err3[i]<-max(abs(P_X-pbf$Bel))
}
boxplot(err1,err2,err3,names=c(100,1000,10000),xlab="sample size",ylab="error")
```