

Test the likelihood calculation integrated across all partitions within a marker group for a given subtype

Preliminaries

```
setwd("/Users/chwu/Documents/research/bfc/github/Forensic-Fluids/")

library(salso)
source("forensicFunctions.R")

load("allParts5And7EltsMatsAndSets.rda")

singleTypeFilePath = "data.csv"
single.df = read.csv(singleTypeFilePath, header = TRUE) # whole single sample data
slv.df = single.df[single.df$Sample == 3,] # saliva
dim(slv.df)

## [1] 81 29

sample.df = slv.df[,-1] # single sample data, delete the index column
dim(sample.df)

## [1] 81 28

checkAbsDiff = function(threshold = 1e-10, expected = NULL, answer = NULL){
  test = abs(expected - answer)
  message(paste("Absolute difference", signif(test, 5), sep = ": "))
  message(paste("Test passed", test < threshold, sep = ": "))
}

checkAbsRelativeDiff = function(threshold = 1e-10, expected = NULL, answer = NULL, rel = NULL){
  test = abs(expected - answer)/rel
  message(paste("Absolute relative difference", signif(test, 5), sep = ": "))
  message(paste("Test passed", test < threshold, sep = ": "))
}
```

Partition with three elements

```
elts3AllPartMat = enumerate.partitions(3)
# [,1] [,2] [,3]
# [1,] 1 1 1
# [2,] 1 1 2
# [3,] 1 2 1
# [4,] 1 2 2
# [5,] 1 2 3

elts3AllPartSet = apply(elts3AllPartMat, 1,
  function(z){
```

```

        setCount = max(z)
        sets = list()
        for(setIndex in 1:setCount){
            sets[[setIndex]] = sort(which(z == setIndex))
        }

        return(sets)
    })

```

```

set.seed(222)
kmeansClust3Index = c(1, 3, 1, 2, 2, 2, 2, 2, 2,
                      3, 3, 2, 3, 2, 3, 2, 2, 1,
                      3, 2, 3, 1, 2, 2, 2, 2, 2,
                      3, 2, 3, 2, 1, 2, 3, 2, 1,
                      2, 1, 1, 2, 2, 2, 2, 2, 2,
                      2, 3, 2, 2, 1, 1, 1, 2, 2,
                      3, 3, 3, 3, 3, 2, 3, 3, 3,
                      3, 2, 2, 2, 2, 3, 3, 1, 2,
                      1, 1, 1, 1, 1, 1, 3, 3, 1)
slvClust1ExIndx = sample(which(kmeansClust3Index==1), 2)
# 180 183
# 75 78
slvClust2ExIndx = sample(which(kmeansClust3Index==2), 5)
# 146 147 145 121 122
# 41 42 40 16 17
slvClust3ExIndx = sample(which(kmeansClust3Index==3), 3)
# 185 162 126
# 80 57 21

rowExIndx = c(slvClust1ExIndx, slvClust2ExIndx, slvClust3ExIndx)
sample.df[rowExIndx ,1:5]

```

```

##      Sample CYP HBD1 Lcris Lgas
## 180      3   0   1     0   0
## 183      3   0   1     0   0
## 146      3   1   0     0   0
## 147      3   0   0     0   0
## 145      3   0   0     0   0
## 121      3   0   0     0   0
## 122      3   0   0     0   0
## 135      3   0   0     0   0
## 167      3   0   0     0   0
## 107      3   0   0     0   0

```

```

colExIndx = c(1,2,5)
slvEx1.df = sample.df[rowExIndx, c("CYP", "HBD1", "MUC4")]
slvEx1.df

```

```

##      CYP HBD1 MUC4
## 180    0    1    1
## 183    0    1    1
## 146    1    0    0
## 147    0    0    0
## 145    0    0    1

```

```
## 121 0 0 1
## 122 0 0 1
## 135 0 0 0
## 167 0 0 0
## 107 0 0 0
```

Test calcLikIntAllPart Function

Partition 1

Partition: (1, 1, 1)

```
prod(dim(slvEx1.df))
```

```
## [1] 30
```

```
sum(slvEx1.df)
```

```
## [1] 8
```

```
prod(dim(slvEx1.df)) - sum(slvEx1.df)
```

```
## [1] 22
```

```
ex1.expt1 = sum(log(1:9)) - sum(log(25:34))
ex1.ans1 = log(calcLikIntAllPart(eltsAllPartSet = list(list(c(1, 2, 3))),
                                sample = slvEx1.df,
                                alphaC = 2,
                                betaC = 3))
```

```
checkAbsDiff(expected = ex1.expt1,
              answer = ex1.ans1)
```

```
## Absolute difference: 3.5527e-15
```

```
## Test passed: TRUE
```

```
checkAbsRelativeDiff(expected = ex1.expt1,
                      answer = ex1.ans1,
                      rel = min(abs(c(ex1.expt1, ex1.ans1))))
```

```
## Absolute relative difference: 1.6922e-16
```

```
## Test passed: TRUE
```

```
ex1.mdp.expt1 = 15/(7 * 13)
ex1.mdp.ans1 = calcMDPColPrior(alpha = 12, setList = list(c(1,1,1)), setCountMax = 3)
```

```
checkAbsDiff(expected = ex1.mdp.expt1,
              answer = ex1.mdp.ans1)
```

```
## Absolute difference: 9.992e-16
```

```
## Test passed: TRUE
```

```
checkAbsRelativeDiff(expected = ex1.mdp.expt1,
                      answer = ex1.mdp.ans1,
                      rel = min(abs(c(ex1.mdp.expt1, ex1.mdp.ans1))))
```

```
## Absolute relative difference: 6.0618e-15
```

```
## Test passed: TRUE
```

Partition 2

Partition: (1, 1, 2)

```
prod(dim(slvEx1.df[, 1:2]))
```

```
## [1] 20
```

```
sum(slvEx1.df[, 1:2])
```

```
## [1] 3
```

```
prod(dim(slvEx1.df[, 1:2])) - sum(slvEx1.df[, 1:2])
```

```
## [1] 17
```

```
length(slvEx1.df[, 3])
```

```
## [1] 10
```

```
sum(slvEx1.df[, 3])
```

```
## [1] 5
```

```
ex1.expt2 = (sum(log(1:4)) - sum(log(20:24))) + (sum(log(1:6)) - sum(log(8:14)))
```

```
ex1.ans2 = log(calcLikIntAllPart(eltsAllPartSet = list(list(c(1, 2), c(3))),  
                                sample = slvEx1.df,  
                                alphaC = 2,  
                                betaC = 3))
```

```
checkAbsDiff(expected = ex1.expt2,  
              answer = ex1.ans2)
```

```
## Absolute difference: 3.5527e-15
```

```
## Test passed: TRUE
```

```
checkAbsRelativeDiff(expected = ex1.expt2,  
                     answer = ex1.ans2,  
                     rel = min(abs(c(ex1.expt2, ex1.ans2))))
```

```
## Absolute relative difference: 1.5893e-16
```

```
## Test passed: TRUE
```

```
ex1.mdp.expt2 = 20/(7 * 13)
```

```
ex1.mdp.ans2 = calcMDPColPrior(alpha = 12, setList = list(c(1,1), c(3)), setCountMax = 3)
```

```
abs(ex1.mdp.expt2 - ex1.mdp.ans2)/max(c(ex1.mdp.expt2, ex1.mdp.ans2))
```

```
## [1] 1.515454e-15
```

```
abs((ex1.mdp.expt2 - ex1.mdp.ans2)/abs(max(c(ex1.mdp.expt2, ex1.mdp.ans2)))) < 1e-10
```

```
## [1] TRUE
```

```
checkAbsDiff(expected = ex1.mdp.expt2,  
              answer = ex1.mdp.ans2)
```

```
## Absolute difference: 3.3307e-16
```

```
## Test passed: TRUE
```

```
checkAbsRelativeDiff(expected = ex1.mdp.expt2,
                      answer = ex1.mdp.ans2,
                      rel = min(abs(c(ex1.mdp.expt2, ex1.mdp.ans2))))
```

```
## Absolute relative difference: 1.5155e-15
```

```
## Test passed: TRUE
```

Partition 3

Partition: (1, 2, 1)

```
prod(dim(slvEx1.df[, c(1,3)]))
```

```
## [1] 20
```

```
sum(slvEx1.df[, c(1,3)])
```

```
## [1] 6
```

```
length(slvEx1.df[, 2]) - sum(slvEx1.df[, 2])
```

```
## [1] 8
```

```
ex1.expt3 = ((sum(log(1:7)) - sum(log(17:24)))) +
  ((sum(log(1:3)) - sum(log(11:14))))
ex1.ans3 = log(calcLikIntAllPart(eltsAllPartSet = list(list(c(1, 3), c(2))),
                              sample = slvEx1.df,
                              alphaC = 2,
                              betaC = 3))
```

```
checkAbsDiff(expected = ex1.expt3,
              answer = ex1.ans3)
```

```
## Absolute difference: 0
```

```
## Test passed: TRUE
```

```
checkAbsRelativeDiff(expected = ex1.expt3,
                      answer = ex1.ans3,
                      rel = min(abs(c(ex1.expt3, ex1.ans3))))
```

```
## Absolute relative difference: 0
```

```
## Test passed: TRUE
```

```
ex1.mdp.expt3 = 20/(7 * 13)
ex1.mdp.ans3 = calcMDPColPrior(alpha = 12,
                              setList = list(c(1,3), c(2)),
                              setCountMax = 3)
```

```
checkAbsDiff(expected = ex1.mdp.expt3,
              answer = ex1.mdp.ans3)
```

```
## Absolute difference: 3.3307e-16
```

```
## Test passed: TRUE
```

```
checkAbsRelativeDiff(expected = ex1.mdp.expt3,
                      answer = ex1.mdp.ans3,
                      rel = min(abs(c(ex1.mdp.expt3, ex1.mdp.ans3))))
```

```
## Absolute relative difference: 1.5155e-15
```

```
## Test passed: TRUE
```

Partition 4

Partition: (1, 2, 2)

```
# 1 2 2
length(slvEx1.df[, 1])
```

```
## [1] 10
```

```
sum(slvEx1.df[, 1])
```

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

```
prod(dim(slvEx1.df[, c(2,3)]))
```

```
## [1] 20
```

```
sum(slvEx1.df[, c(2,3)])
```

```
## [1] 7
```

```
ex1.expt4 = -sum(log(c(6, 13, 14))) +
  ((sum(log(1:8)) - sum(log(16:24))))
ex1.ans4 = log(calcLikIntAllPart(eltsAllPartSet = list(list(c(1), c(2, 3))),
  sample = slvEx1.df,
  alphaC = 2,
  betaC = 3))
```

```
checkAbsDiff(expected = ex1.expt4,
  answer = ex1.ans4)
```

```
## Absolute difference: 0
```

```
## Test passed: TRUE
```

```
checkAbsRelativeDiff(expected = ex1.expt4,
  answer = ex1.ans4,
  rel = min(abs(c(ex1.expt4, ex1.ans4))))
```

```
## Absolute relative difference: 0
```

```
## Test passed: TRUE
```

```
ex1.mdp.expt4 = 20/(7 * 13)
ex1.mdp.ans4 = calcMDPColPrior(alpha = 12,
  setList = list(c(2,3), c(1)),
  setCountMax = 3)
```

```
checkAbsDiff(expected = ex1.mdp.expt4,
  answer = ex1.mdp.ans4)
```

```
## Absolute difference: 3.3307e-16
```

```
## Test passed: TRUE
```

```
checkAbsRelativeDiff(expected = ex1.mdp.expt4,
  answer = ex1.mdp.ans4,
  rel = min(abs(c(ex1.mdp.expt4, ex1.mdp.ans4))))
```

```
## Absolute relative difference: 1.5155e-15
```

```
## Test passed: TRUE
```

Partition 5

Partition: (1, 2, 3)

```
length(slvEx1.df[, 1])
```

```
## [1] 10
```

```
sum(slvEx1.df[, 1])
```

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

```
length(slvEx1.df[, 2])
```

```
## [1] 10
```

```
sum(slvEx1.df[, 2])
```

```
## [1] 2
```

```
length(slvEx1.df[, 3])
```

```
## [1] 10
```

```
sum(slvEx1.df[, 3])
```

```
## [1] 5
```

```
ex1.expt5 = (-sum(log(c(6, 13, 14)))) +  
  -sum(log(c(2, 11, 13, 14))) +  
  (sum(log(1:6)) - sum(log(c(8:14))))  
ex1.ans5 = log(calcLikIntAllPart(eltsAllPartSet = list(list(c(1), c(2), c(3))),  
                                sample = slvEx1.df,  
                                alphaC = 2,  
                                betaC = 3))
```

```
checkAbsDiff(expected = ex1.expt5,  
              answer = ex1.ans5)
```

```
## Absolute difference: 3.5527e-15
```

```
## Test passed: TRUE
```

```
checkAbsRelativeDiff(expected = ex1.expt5,  
                      answer = ex1.ans5,  
                      rel = min(abs(c(ex1.expt5, ex1.ans5))))
```

```
## Absolute relative difference: 1.3999e-16
```

```
## Test passed: TRUE
```

```
ex1.mdp.expt5 = 16/(7 * 13)
```

```
ex1.mdp.ans5 = calcMDPColPrior(alpha = 12,  
                               setList = list(c(1), c(2), c(3)),  
                               setCountMax = 3)
```

```
checkAbsDiff(expected = ex1.mdp.expt5,  
              answer = ex1.mdp.ans5)
```

```
## Absolute difference: 4.1633e-16
```

```
## Test passed: TRUE
```

```
checkAbsRelativeDiff(expected = ex1.mdp.expt5,  
                      answer = ex1.mdp.ans5,  
                      rel = min(abs(c(ex1.mdp.expt5, ex1.mdp.ans5))))
```

```
## Absolute relative difference: 2.3679e-15
```

```
## Test passed: TRUE
```

Test calcLogSubTypeLik Function

```
ex1.subtype.lik.expt = ex1.mdp.expt1*exp(ex1.expt1) +  
  ex1.mdp.expt2*exp(ex1.expt2) +  
  ex1.mdp.expt3*exp(ex1.expt3) +  
  ex1.mdp.expt4*exp(ex1.expt4) +  
  ex1.mdp.expt5*exp(ex1.expt5)
```

```
ex1.subtype.log.lik.expt = log(ex1.subtype.lik.expt)
```

```
elts3AllPartSetMDPPrior = unlist(lapply(elts3AllPartSet,  
                                       calcMDPColPrior,  
                                       alpha = 12,  
                                       setCountMax = 3))
```

```
ex1.subtype.log.lik.ans = calcLogSubTypeLik(eltsAllPartSetList = list(elts3AllPartSet),  
                                           eltsAllPartSetPriorList = list(elts3AllPartSetMDPPrior),  
                                           sample = slvEx1.df,  
                                           mkrList = list(colnames(slvEx1.df)),  
                                           alphaC = 2,  
                                           betaC = 3)
```

```
checkAbsDiff(expected = ex1.subtype.log.lik.expt,  
             answer = ex1.subtype.log.lik.ans)
```

```
## Absolute difference: 7.1054e-15
```

```
## Test passed: TRUE
```

```
checkAbsRelativeDiff(expected = ex1.subtype.log.lik.expt,  
                      answer = ex1.subtype.log.lik.ans,  
                      rel = min(abs(c(ex1.subtype.log.lik.expt,  
                                       ex1.subtype.log.lik.ans))))
```

```
## Absolute relative difference: 3.1794e-16
```

```
## Test passed: TRUE
```

Partition with four elements

```
elts4AllPartMat = enumerate.partitions(4)  
#      [,1] [,2] [,3] [,4]  
# [1,]    1    1    1    1  
# [2,]    1    1    1    2  
# [3,]    1    1    2    1
```



```

# [4,] 1 1 2 2
# [5,] 1 1 2 3
# [6,] 1 2 1 1
# [7,] 1 2 1 2
# [8,] 1 2 1 3
# [9,] 1 2 2 1
# [10,] 1 2 2 2
# [11,] 1 2 2 3
# [12,] 1 2 3 1
# [13,] 1 2 3 2
# [14,] 1 2 3 3
# [15,] 1 2 3 4

elts4AllPartSet = apply(elts4AllPartMat, 1,
  function(z){
    setCount = max(z)
    sets = list()
    for(setIndex in 1:setCount){
      sets[[setIndex]] = sort(which(z == setIndex))
    }

    return(sets)
  })

```

```

set.seed(123)
slvClust1ExIndx2 = sample(which(kmeansClust3Index==1), 2)
# 180 179
# 75 74
slvClust2ExIndx2 = sample(which(kmeansClust3Index==2), 5)
# 111 173 130 148 149
# 6 68 25 43 44
slvClust3ExIndx2 = sample(which(kmeansClust3Index==3), 3)
# 120 168 135
# 15 63 30

```

```

rowExIndx2 = c(slvClust1ExIndx2, slvClust2ExIndx2, slvClust3ExIndx2)
slvEx2.df = sample.df[rowExIndx2 ,c("HTN3", "MUC7", "PRB4", "SMR3B" )]
slvEx2.df

```

##	HTN3	MUC7	PRB4	SMR3B
## 180	1	1	1	1
## 179	1	1	1	1
## 111	1	1	0	1
## 173	1	1	0	1
## 130	1	1	0	1
## 148	1	1	0	1
## 149	1	1	0	1
## 120	0	1	0	1
## 168	0	1	0	0
## 135	0	1	0	1

Test Function

Partition 1

```
sum(slvEx2.df)

## [1] 28

prod(dim(slvEx2.df))

## [1] 40

ex2.expt1 = sum(log(1:13)) - sum(log(31:44))
ex2.ans1 = log(calcLikIntAllPart(eltsAllPartSet = list(list(c(1:4))),
                                sample = slvEx2.df,
                                alphaC = 3,
                                betaC = 2))

checkAbsDiff(expected = ex2.expt1,
              answer = ex2.ans1)

## Absolute difference: 0
## Test passed: TRUE

checkAbsRelativeDiff(expected = ex2.expt1,
                     answer = ex2.ans1,
                     rel = min(abs(c(ex2.expt1, ex2.ans1))))

## Absolute relative difference: 0
## Test passed: TRUE

ex2.mdp.expt1 = 35/(3 * 17 * 19)
ex2.mdp.ans1 = calcMDPColPrior(alpha = 16,
                               setList = list(c(1, 2, 3, 4)),
                               setCountMax = 4)

checkAbsDiff(expected = ex2.mdp.expt1,
              answer = ex2.mdp.ans1)

## Absolute difference: 1.0408e-16
## Test passed: TRUE

checkAbsRelativeDiff(expected = ex2.mdp.expt1,
                     answer = ex2.mdp.ans1,
                     rel = min(abs(c(ex2.mdp.expt1, ex2.mdp.ans1))))

## Absolute relative difference: 2.8816e-15
## Test passed: TRUE
```

Partition 2

Partition pattern: (1, 1, 1, 2)

```
sum(slvEx2.df[, 1:3])
```

```
## [1] 19
```

```

prod(dim(slvEx2.df[, 1:3]))

## [1] 30
sum(slvEx2.df[,4])

## [1] 9
length(slvEx2.df[,4])

## [1] 10
ex2.expt2.clust1.log.lik = sum(log(1:12)) - sum(log(22:34))
ex2.expt2.clust2.log.lik = -(log(6) + log(13) + log(14))

ex2.expt2 = ex2.expt2.clust1.log.lik + ex2.expt2.clust2.log.lik
ex2.ans2 = log(calcLikIntAllPart(eltsAllPartSet = list(list(c(1:3), c(4))),
                                sample = slvEx2.df,
                                alphaC = 3,
                                betaC = 2))

checkAbsDiff(expected = ex2.expt2,
              answer = ex2.ans2)

## Absolute difference: 3.5527e-15
## Test passed: TRUE
checkAbsRelativeDiff(expected = ex2.expt2,
                     answer = ex2.ans2,
                     rel = min(abs(c(ex2.expt2, ex2.ans2))))

## Absolute relative difference: 1.176e-16
## Test passed: TRUE
ex2.mdp.expt2 = 20/(17 * 19)
ex2.mdp.ans2 = calcMDPColPrior(alpha = 16,
                              setList = list(c(1, 2, 3), c(4)),
                              setCountMax = 4)

checkAbsDiff(expected = ex2.mdp.expt2,
              answer = ex2.mdp.ans2)

## Absolute difference: 3.5388e-16
## Test passed: TRUE
checkAbsRelativeDiff(expected = ex2.mdp.expt2,
                     answer = ex2.mdp.ans2,
                     rel = min(abs(c(ex2.mdp.expt2, ex2.mdp.ans2))))

## Absolute relative difference: 5.7152e-15
## Test passed: TRUE

```

Partition 3

Partition pattern: (1, 1, 2, 1)

```

sum(slvEx2.df[, c(1, 2, 4)])

## [1] 26
prod(dim(slvEx2.df[, c(1, 2, 4)]))

## [1] 30
sum(slvEx2.df[,3])

## [1] 2
length(slvEx2.df[,3])

## [1] 10
ex2.expt3.clust1.log.lik = sum(log(1:5)) - sum(log(29:34))
ex2.expt3.clust2.log.lik = -(log(5) + log(11) + log(13) + log(14))

ex2.expt3 = ex2.expt3.clust1.log.lik + ex2.expt3.clust2.log.lik
ex2.ans3 = log(calcLikIntAllPart(eltsAllPartSet = list(list(c(1,2,4), c(3))),
                                sample = slvEx2.df,
                                alphaC = 3,
                                betaC = 2))

checkAbsDiff(expected = ex2.expt3,
              answer = ex2.ans3)

## Absolute difference: 3.5527e-15
## Test passed: TRUE
checkAbsRelativeDiff(expected = ex2.expt3,
                      answer = ex2.ans3,
                      rel = min(abs(c(ex2.expt3, ex2.ans3))))

## Absolute relative difference: 1.4146e-16
## Test passed: TRUE
ex2.mdp.expt3 = 20/(17 * 19)
ex2.mdp.ans3 = calcMDPColPrior(alpha = 16,
                               setList = list(c(1, 2, 4), c(3)),
                               setCountMax = 4)

checkAbsDiff(expected = ex2.mdp.expt3,
              answer = ex2.mdp.ans3)

## Absolute difference: 3.5388e-16
## Test passed: TRUE
checkAbsRelativeDiff(expected = ex2.mdp.expt3,
                      answer = ex2.mdp.ans3,
                      rel = min(abs(c(ex2.mdp.expt3, ex2.mdp.ans3))))

## Absolute relative difference: 5.7152e-15
## Test passed: TRUE

```

Partition 4

Partition pattern: (1, 1, 2, 2)

```
sum(slvEx2.df[, c(1, 2)])
```

```
## [1] 17
```

```
prod(dim(slvEx2.df[, c(1, 2)]))
```

```
## [1] 20
```

```
sum(slvEx2.df[, c(3, 4)])
```

```
## [1] 11
```

```
prod(dim(slvEx2.df[, c(3, 4)]))
```

```
## [1] 20
```

```
ex2.expt4.clust1.log.lik = - sum(log(20:23))
```

```
ex2.expt4.clust2.log.lik = sum(log(1:10)) - sum(log(14:24))
```

```
ex2.expt4 = ex2.expt4.clust1.log.lik + ex2.expt4.clust2.log.lik
```

```
ex2.ans4 = log(calcLikIntAllPart(eltsAllPartSet = list(list(c(1,2), c(3,4))),  
                                sample = slvEx2.df,  
                                alphaC = 3,  
                                betaC = 2))
```

```
checkAbsDiff(expected = ex2.expt4,  
              answer = ex2.ans4)
```

```
## Absolute difference: 0
```

```
## Test passed: TRUE
```

```
checkAbsRelativeDiff(expected = ex2.expt4,  
                      answer = ex2.ans4,  
                      rel = min(abs(c(ex2.expt4, ex2.ans4))))
```

```
## Absolute relative difference: 0
```

```
## Test passed: TRUE
```

```
ex2.mdp.expt4 = 50/(3 * 17 * 19)
```

```
ex2.mdp.ans4 = calcMDPColPrior(alpha = 16,  
                               setList = list(c(1, 2), c(3, 4)),  
                               setCountMax = 4)
```

```
checkAbsDiff(expected = ex2.mdp.expt4,  
              answer = ex2.mdp.ans4)
```

```
## Absolute difference: 1.0408e-16
```

```
## Test passed: TRUE
```

```
checkAbsRelativeDiff(expected = ex2.mdp.expt4,  
                      answer = ex2.mdp.ans4,  
                      rel = min(abs(c(ex2.mdp.expt4, ex2.mdp.ans4))))
```

```
## Absolute relative difference: 2.0171e-15
```

```
## Test passed: TRUE
```

Partition 5

Partition pattern: (1, 1, 2, 3)

```
sum(slvEx2.df[, c(1, 2)])
```

```
## [1] 17
```

```
prod(dim(slvEx2.df[, c(1, 2)]))
```

```
## [1] 20
```

```
sum(slvEx2.df[, 3])
```

```
## [1] 2
```

```
length(slvEx2.df[, 3])
```

```
## [1] 10
```

```
sum(slvEx2.df[, 4])
```

```
## [1] 9
```

```
length(slvEx2.df[,4])
```

```
## [1] 10
```

```
ex2.expt5.clust1.log.lik = - sum(log(20:23))
ex2.expt5.clust2.log.lik = - sum(log(c(5, 11, 13, 14)))
ex2.expt5.clust3.log.lik = - sum(log(c(6, 13, 14)))
```

```
ex2.expt5 = ex2.expt5.clust1.log.lik + ex2.expt5.clust2.log.lik + ex2.expt5.clust3.log.lik
ex2.ans5 = log(calcLikIntAllPart(eltsAllPartSet = list(list(c(1,2), c(3), c(4))),
                                sample = slvEx2.df,
                                alphaC = 3,
                                betaC = 2))
```

```
checkAbsDiff(expected = ex2.expt5,
              answer = ex2.ans5)
```

```
## Absolute difference: 3.5527e-15
```

```
## Test passed: TRUE
```

```
checkAbsRelativeDiff(expected = ex2.expt5,
                     answer = ex2.ans5,
                     rel = min(abs(c(ex2.expt5, ex2.ans5))))
```

```
## Absolute relative difference: 1.2477e-16
```

```
## Test passed: TRUE
```

```
ex2.mdp.expt5 = 80/(3 * 17 * 19)
ex2.mdp.ans5 = calcMDPColPrior(alpha = 16,
                              setList = list(c(1, 2), c(3), c(4)),
                              setCountMax = 4)
```

```
checkAbsDiff(expected = ex2.mdp.expt5,
              answer = ex2.mdp.ans5)
```

```
## Absolute difference: 1.9429e-16
```

```
## Test passed: TRUE
```

```
checkAbsRelativeDiff(expected = ex2.mdp.expt5,  
                      answer = ex2.mdp.ans5,  
                      rel = min(abs(c(ex2.mdp.expt5, ex2.mdp.ans5))))
```

```
## Absolute relative difference: 2.3533e-15
```

```
## Test passed: TRUE
```

Partition 6

Partition pattern: (1, 2, 1, 1)

```
sum(slvEx2.df[, c(1, 3, 4)])
```

```
## [1] 18
```

```
prod(dim(slvEx2.df[, c(1, 3, 4)]))
```

```
## [1] 30
```

```
sum(slvEx2.df[, 2])
```

```
## [1] 10
```

```
length(slvEx2.df[, 2])
```

```
## [1] 10
```

```
ex2.expt6.clust1.log.lik = sum(log(1:13)) - sum(log(21:34))
```

```
ex2.expt6.clust2.log.lik = -(log(13) + log(14))
```

```
ex2.expt6 = ex2.expt6.clust1.log.lik + ex2.expt6.clust2.log.lik
```

```
ex2.ans6 = log(calcLikIntAllPart(eltsAllPartSet = list(list(c(1,3,4), c(2))),  
                                sample = slvEx2.df,  
                                alphaC = 3,  
                                betaC = 2))
```

```
checkAbsDiff(expected = ex2.expt6,  
              answer = ex2.ans6)
```

```
## Absolute difference: 0
```

```
## Test passed: TRUE
```

```
checkAbsRelativeDiff(expected = ex2.expt6,  
                      answer = ex2.ans6,  
                      rel = min(abs(c(ex2.expt6, ex2.ans6))))
```

```
## Absolute relative difference: 0
```

```
## Test passed: TRUE
```

```
ex2.mdp.expt6 = 20/(17 * 19)
```

```
ex2.mdp.ans6 = calcMDPColPrior(alpha = 16,  
                               setList = list(c(1, 3, 4), c(2)),  
                               setCountMax = 4)
```

```
checkAbsDiff(expected = ex2.mdp.expt6,  
             answer = ex2.mdp.ans6)
```

```
## Absolute difference: 3.5388e-16
```

```
## Test passed: TRUE
```

```
checkAbsRelativeDiff(expected = ex2.mdp.expt6,  
                    answer = ex2.mdp.ans6,  
                    rel = min(abs(c(ex2.mdp.expt6, ex2.mdp.ans6))))
```

```
## Absolute relative difference: 5.7152e-15
```

```
## Test passed: TRUE
```

Partition 7

Partition: (1, 2, 1, 2)

```
sum(slvEx2.df[, c(1, 3)])
```

```
## [1] 9
```

```
prod(dim(slvEx2.df[, c(1, 3)]))
```

```
## [1] 20
```

```
sum(slvEx2.df[, c(2, 4)])
```

```
## [1] 19
```

```
prod(dim(slvEx2.df[, c(2, 4)]))
```

```
## [1] 20
```

```
ex2.expt7.clust1.log.lik = sum(log(1:11)) - sum(log(13:24))  
ex2.expt7.clust2.log.lik = -sum(log(c(11, 23, 24)))
```

```
ex2.expt7 = ex2.expt7.clust1.log.lik + ex2.expt7.clust2.log.lik  
ex2.ans7 = log(calcLikIntAllPart(eltsAllPartSet = list(list(c(1,3), c(2,4))),  
                               sample = slvEx2.df,  
                               alphaC = 3,  
                               betaC = 2))
```

```
checkAbsDiff(expected = ex2.expt7,  
             answer = ex2.ans7)
```

```
## Absolute difference: 7.1054e-15
```

```
## Test passed: TRUE
```

```
checkAbsRelativeDiff(expected = ex2.expt7,  
                    answer = ex2.ans7,  
                    rel = min(abs(c(ex2.expt7, ex2.ans7))))
```

```
## Absolute relative difference: 2.7322e-16
```

```
## Test passed: TRUE
```

```
ex2.mdp.expt7 = 50/(3 * 17 * 19)  
ex2.mdp.ans7 = calcMDPColPrior(alpha = 16,  
                              setList = list(c(1, 3), c(2, 4)),
```



```

                                setCountMax = 4)

checkAbsDiff(expected = ex2.mdp.expt7,
              answer = ex2.mdp.ans7)

## Absolute difference: 1.0408e-16
## Test passed: TRUE

checkAbsRelativeDiff(expected = ex2.mdp.expt7,
                      answer = ex2.mdp.ans7,
                      rel = min(abs(c(ex2.mdp.expt7, ex2.mdp.ans7))))

## Absolute relative difference: 2.0171e-15
## Test passed: TRUE

Partition 8

Partition pattern: (1, 2, 1, 3)
sum(slvEx2.df[, c(1, 3)])

## [1] 9
prod(dim(slvEx2.df[, c(1, 3)]))

## [1] 20
sum(slvEx2.df[, 2])

## [1] 10
length(slvEx2.df[, 2])

## [1] 10
sum(slvEx2.df[, 4])

## [1] 9
length(slvEx2.df[,4])

## [1] 10
ex2.expt8.clust1.log.lik = sum(log(1:11)) - sum(log(13:24))
ex2.expt8.clust2.log.lik = - (log(13) + log(14))
ex2.expt8.clust3.log.lik = - sum(log(c(6, 13, 14)))

ex2.expt8 = ex2.expt8.clust1.log.lik + ex2.expt8.clust2.log.lik + ex2.expt8.clust3.log.lik
ex2.ans8 = log(calcLikIntAllPart(eltsAllPartSet = list(list(c(1,3), c(2), c(4))),
                                sample = slvEx2.df,
                                alphaC = 3,
                                betaC = 2))

checkAbsDiff(expected = ex2.expt8,
              answer = ex2.ans8)

## Absolute difference: 3.5527e-15
## Test passed: TRUE

```

```
checkAbsRelativeDiff(expected = ex2.expt8,
                      answer = ex2.ans8,
                      rel = min(abs(c(ex2.expt8, ex2.ans8))))
```

```
## Absolute relative difference: 1.2045e-16
```

```
## Test passed: TRUE
```

```
ex2.mdp.expt8 = 80/(3 * 17 * 19)
ex2.mdp.ans8 = calcMDPColPrior(alpha = 16,
                              setList = list(c(1, 3), c(2), c(4)),
                              setCountMax = 4)
```

```
checkAbsDiff(expected = ex2.mdp.expt8,
              answer = ex2.mdp.ans8)
```

```
## Absolute difference: 1.9429e-16
```

```
## Test passed: TRUE
```

```
checkAbsRelativeDiff(expected = ex2.mdp.expt8,
                      answer = ex2.mdp.ans8,
                      rel = min(abs(c(ex2.mdp.expt8, ex2.mdp.ans8))))
```

```
## Absolute relative difference: 2.3533e-15
```

```
## Test passed: TRUE
```

Partition 9

```
Partition: (1, 2, 2, 1)
```

```
sum(slvEx2.df[, c(1, 4)])
```

```
## [1] 16
```

```
prod(dim(slvEx2.df[, c(1, 4)]))
```

```
## [1] 20
```

```
sum(slvEx2.df[, c(2, 3)])
```

```
## [1] 12
```

```
prod(dim(slvEx2.df[, c(2, 3)]))
```

```
## [1] 20
```

```
ex2.expt9.clust1.log.lik = sum(log(1:5)) - sum(log(19:24))
ex2.expt9.clust2.log.lik = sum(log(1:9)) - sum(log(15:24))
```

```
ex2.expt9 = ex2.expt9.clust1.log.lik + ex2.expt9.clust2.log.lik
ex2.ans9 = log(calcLikIntAllPart(eltsAllPartSet = list(list(c(1,4), c(2,3))),
                                sample = slvEx2.df,
                                alphaC = 3,
                                betaC = 2))
```

```
checkAbsDiff(expected = ex2.expt9,
              answer = ex2.ans9)
```

```
## Absolute difference: 0
```

```
## Test passed: TRUE
checkAbsRelativeDiff(expected = ex2.expt9,
                      answer = ex2.ans9,
                      rel = min(abs(c(ex2.expt9, ex2.ans9))))

## Absolute relative difference: 0
## Test passed: TRUE
ex2.mdp.expt9 = 50/(3 * 17 * 19)
ex2.mdp.ans9 = calcMDPColPrior(alpha = 16,
                              setList = list(c(1, 4), c(2, 3)),
                              setCountMax = 4)

checkAbsDiff(expected = ex2.mdp.expt9,
              answer = ex2.mdp.ans9)

## Absolute difference: 1.0408e-16
## Test passed: TRUE
checkAbsRelativeDiff(expected = ex2.mdp.expt9,
                      answer = ex2.mdp.ans9,
                      rel = min(abs(c(ex2.mdp.expt9, ex2.mdp.ans9))))

## Absolute relative difference: 2.0171e-15
## Test passed: TRUE
```

Partition 10

Partition: (1, 2, 2, 2)

```
sum(slvEx2.df[, 1])

## [1] 7
length(slvEx2.df[, 1])

## [1] 10
sum(slvEx2.df[, 2:4])

## [1] 21
prod(dim(slvEx2.df[, 2:4]))

## [1] 30
ex2.expt10.clust1.log.lik = -sum(log(5) + log(11) + log(13) + log(14))
ex2.expt10.clust2.log.lik = sum(log(1:10)) - sum(log(24:34))

ex2.expt10 = ex2.expt10.clust1.log.lik + ex2.expt10.clust2.log.lik
ex2.ans10 = log(calcLikIntAllPart(eltsAllPartSet = list(list(c(1), c(2:4))),
                                sample = slvEx2.df,
                                alphaC = 3,
                                betaC = 2))

checkAbsDiff(expected = ex2.expt10,
              answer = ex2.ans10)
```

```
## Absolute difference: 3.5527e-15
```

```
## Test passed: TRUE
```

```
checkAbsRelativeDiff(expected = ex2.expt10,  
                      answer = ex2.ans10,  
                      rel = min(abs(c(ex2.expt10, ex2.ans10))))
```

```
## Absolute relative difference: 1.143e-16
```

```
## Test passed: TRUE
```

```
ex2.mdp.expt10 = 20/(17 * 19)  
ex2.mdp.ans10 = calcMDPColPrior(alpha = 16,  
                                setList = list(c(1), c(2, 3, 4)),  
                                setCountMax = 4)
```

```
checkAbsDiff(expected = ex2.mdp.expt10,  
             answer = ex2.mdp.ans10)
```

```
## Absolute difference: 3.5388e-16
```

```
## Test passed: TRUE
```

```
checkAbsRelativeDiff(expected = ex2.mdp.expt10,  
                      answer = ex2.mdp.ans10,  
                      rel = min(abs(c(ex2.mdp.expt10, ex2.mdp.ans10))))
```

```
## Absolute relative difference: 5.7152e-15
```

```
## Test passed: TRUE
```

Partition 11

Partition: (1, 2, 2, 3)

```
sum(slvEx2.df[, 1])
```

```
## [1] 7
```

```
length(slvEx2.df[, 1])
```

```
## [1] 10
```

```
sum(slvEx2.df[, c(2, 3)])
```

```
## [1] 12
```

```
prod(dim(slvEx2.df[, c(2, 3)]))
```

```
## [1] 20
```

```
sum(slvEx2.df[, 4])
```

```
## [1] 9
```

```
length(slvEx2.df[,4])
```

```
## [1] 10
```

```
ex2.expt11.clust1.log.lik = -sum(log(c(5, 11, 13, 14)))  
ex2.expt11.clust2.log.lik = sum(log(1:9)) - sum(log(15:24))  
ex2.expt11.clust3.log.lik = -sum(log(c(6, 13, 14)))
```

```

ex2.expt11 = ex2.expt11.clust1.log.lik +
  ex2.expt11.clust2.log.lik +
  ex2.expt11.clust3.log.lik
ex2.ans11 = log(calcLikIntAllPart(eltsAllPartSet = list(list(c(1), c(2, 3), c(4))),
  sample = slvEx2.df,
  alphaC = 3,
  betaC = 2))

checkAbsDiff(expected = ex2.expt11,
  answer = ex2.ans11)

```

Absolute difference: 0

Test passed: TRUE

```

checkAbsRelativeDiff(expected = ex2.expt11,
  answer = ex2.ans11,
  rel = min(abs(c(ex2.expt11, ex2.ans11))))

```

Absolute relative difference: 0

Test passed: TRUE

```

ex2.mdp.expt11 = 80/(3 * 17 * 19)
ex2.mdp.ans11 = calcMDPColPrior(alpha = 16,
  setList = list(c(1), c(2, 3), c(4)),
  setCountMax = 4)

```

```

checkAbsDiff(expected = ex2.mdp.expt11,
  answer = ex2.mdp.ans11)

```

Absolute difference: 1.9429e-16

Test passed: TRUE

```

checkAbsRelativeDiff(expected = ex2.mdp.expt11,
  answer = ex2.mdp.ans11,
  rel = min(abs(c(ex2.mdp.expt11, ex2.mdp.ans11))))

```

Absolute relative difference: 2.3533e-15

Test passed: TRUE

Partition 12

Partition: (1, 2, 3, 1)

```
sum(slvEx2.df[, c(1, 4)])
```

[1] 16

```
prod(dim(slvEx2.df[, c(1, 4)]))
```

[1] 20

```
sum(slvEx2.df[, 2])
```

[1] 10

```
length(slvEx2.df[, 2])
```

```

## [1] 10
sum(slvEx2.df[, 3])

## [1] 2
length(slvEx2.df[, 3])

## [1] 10
ex2.expt12.clust1.log.lik = sum(log(1:5)) - sum(log(19:24))
ex2.expt12.clust2.log.lik = - sum(log(13) + log(14))
ex2.expt12.clust3.log.lik = -sum(log(c(5, 11, 13, 14)))

ex2.expt12 = ex2.expt12.clust1.log.lik +
  ex2.expt12.clust2.log.lik +
  ex2.expt12.clust3.log.lik
ex2.ans12 = log(calcLikIntAllPart(eltsAllPartSet = list(list(c(1, 4), c(2), c(3))),
  sample = slvEx2.df,
  alphaC = 3,
  betaC = 2))

checkAbsDiff(expected = ex2.expt12,
  answer = ex2.ans12)

## Absolute difference: 0
## Test passed: TRUE
checkAbsRelativeDiff(expected = ex2.expt12,
  answer = ex2.ans12,
  rel = min(abs(c(ex2.expt12, ex2.ans12))))

## Absolute relative difference: 0
## Test passed: TRUE
ex2.mdp.expt12 = 80/(3 * 17 * 19)
ex2.mdp.ans12 = calcMDPColPrior(alpha = 16,
  setList = list(c(1, 4), c(2), c(3)),
  setCountMax = 4)

checkAbsDiff(expected = ex2.mdp.expt12,
  answer = ex2.mdp.ans12)

## Absolute difference: 1.9429e-16
## Test passed: TRUE
checkAbsRelativeDiff(expected = ex2.mdp.expt12,
  answer = ex2.mdp.ans12,
  rel = min(abs(c(ex2.mdp.expt12, ex2.mdp.ans12))))

## Absolute relative difference: 2.3533e-15
## Test passed: TRUE

```

Partition 13

Partition: (1, 2, 3, 2)

```

sum(slvEx2.df[, 1])

## [1] 7
length(slvEx2.df[, 1])

## [1] 10
sum(slvEx2.df[, c(2, 4)])

## [1] 19
prod(dim(slvEx2.df[, c(2, 4)]))

## [1] 20
sum(slvEx2.df[, 3])

## [1] 2
length(slvEx2.df[, 3])

## [1] 10
ex2.expt13.clust1.log.lik = -sum(log(c(5, 11, 13, 14)))
ex2.expt13.clust2.log.lik = - (log(11) + log(23) + log(24))
ex2.expt13.clust3.log.lik = -sum(log(c(5, 11, 13, 14)))

ex2.expt13 = ex2.expt13.clust1.log.lik +
  ex2.expt13.clust2.log.lik +
  ex2.expt13.clust3.log.lik
ex2.ans13 = log(calcLikIntAllPart(eltsAllPartSet = list(list(c(1), c(2, 4), c(3))),
                                sample = slvEx2.df,
                                alphaC = 3,
                                betaC = 2))

checkAbsDiff(expected = ex2.expt13,
              answer = ex2.ans13)

## Absolute difference: 0
## Test passed: TRUE
checkAbsRelativeDiff(expected = ex2.expt13,
                     answer = ex2.ans13,
                     rel = min(abs(c(ex2.expt13, ex2.ans13))))

## Absolute relative difference: 0
## Test passed: TRUE
ex2.mdp.expt13 = 80/(3 * 17 * 19)
ex2.mdp.ans13 = calcMDPColPrior(alpha = 16,
                                setList = list(c(1), c(2, 4), c(3)),
                                setCountMax = 4)

checkAbsDiff(expected = ex2.mdp.expt13,
              answer = ex2.mdp.ans13)

## Absolute difference: 1.9429e-16

```

```
## Test passed: TRUE
checkAbsRelativeDiff(expected = ex2.mdp.expt13,
                      answer = ex2.mdp.ans13,
                      rel = min(abs(c(ex2.mdp.expt13, ex2.mdp.ans13))))
```

```
## Absolute relative difference: 2.3533e-15
```

```
## Test passed: TRUE
```

Partition 14

Partition: (1, 2, 3, 3)

```
sum(slvEx2.df[, 1])
```

```
## [1] 7
```

```
length(slvEx2.df[, 1])
```

```
## [1] 10
```

```
sum(slvEx2.df[, 2])
```

```
## [1] 10
```

```
length(slvEx2.df[, 2])
```

```
## [1] 10
```

```
sum(slvEx2.df[, c(3, 4)])
```

```
## [1] 11
```

```
prod(dim(slvEx2.df[, c(3, 4)]))
```

```
## [1] 20
```

```
ex2.expt14.clust1.log.lik = -sum(log(c(5, 11, 13, 14)))
ex2.expt14.clust2.log.lik = - (log(13) + log(14))
ex2.expt14.clust3.log.lik = sum(log(1:10)) -sum(log(c(14:24)))
```

```
ex2.expt14 = ex2.expt14.clust1.log.lik +
  ex2.expt14.clust2.log.lik +
  ex2.expt14.clust3.log.lik
ex2.ans14 = log(calcLikIntAllPart(eltsAllPartSet = list(list(c(1), c(2), c(3, 4))),
                                sample = slvEx2.df,
                                alphaC = 3,
                                betaC = 2))
```

```
checkAbsDiff(expected = ex2.expt14,
              answer = ex2.ans14)
```

```
## Absolute difference: 3.5527e-15
```

```
## Test passed: TRUE
```

```
checkAbsRelativeDiff(expected = ex2.expt14,
                      answer = ex2.ans14,
                      rel = min(abs(c(ex2.expt14, ex2.ans14))))
```

```
## Absolute relative difference: 1.1263e-16
```



```
## Test passed: TRUE
ex2.mdp.expt14 = 80/(3 * 17 * 19)
ex2.mdp.ans14 = calcMDPColPrior(alpha = 16,
                                setList = list(c(1), c(2), c(3, 4)),
                                setCountMax = 4)

checkAbsDiff(expected = ex2.mdp.expt14,
              answer = ex2.mdp.ans14)

## Absolute difference: 1.9429e-16
## Test passed: TRUE
checkAbsRelativeDiff(expected = ex2.mdp.expt14,
                     answer = ex2.mdp.ans14,
                     rel = min(abs(c(ex2.mdp.expt14, ex2.mdp.ans14))))

## Absolute relative difference: 2.3533e-15
## Test passed: TRUE
```

Partition 15

Partition: (1, 2, 3, 4)

```
sum(slvEx2.df[, 1])
```

```
## [1] 7
```

```
length(slvEx2.df[, 1])
```

```
## [1] 10
```

```
sum(slvEx2.df[, 2])
```

```
## [1] 10
```

```
length(slvEx2.df[, 2])
```

```
## [1] 10
```

```
sum(slvEx2.df[, 3])
```

```
## [1] 2
```

```
length(slvEx2.df[, 3])
```

```
## [1] 10
```

```
sum(slvEx2.df[, 4])
```

```
## [1] 9
```

```
length(slvEx2.df[, 4])
```

```
## [1] 10
```

```
ex2.expt15.clust1.log.lik = -sum(log(c(5, 11, 13, 14)))
ex2.expt15.clust2.log.lik = - (log(13) + log(14))
ex2.expt15.clust3.log.lik = -sum(log(c(5, 11, 13, 14)))
ex2.expt15.clust4.log.lik = -sum(log(c(6, 13, 14)))
```

```

ex2.expt15 = ex2.expt15.clust1.log.lik +
  ex2.expt15.clust2.log.lik +
  ex2.expt15.clust3.log.lik +
  ex2.expt15.clust4.log.lik
ex2.ans15 = log(calcLikIntAllPart(eltsAllPartSet = list(list(c(1), c(2), c(3), c(4))),
                                sample = slvEx2.df,
                                alphaC = 3,
                                betaC = 2))

checkAbsDiff(expected = ex2.expt15,
              answer = ex2.ans15)

```

Absolute difference: 3.5527e-15

Test passed: TRUE

```

checkAbsRelativeDiff(expected = ex2.expt15,
                     answer = ex2.ans15,
                     rel = min(abs(c(ex2.expt15, ex2.ans15))))

```

Absolute relative difference: 1.1602e-16

Test passed: TRUE

```

ex2.mdp.expt15 = 64/(3 * 17 * 19)
ex2.mdp.ans15 = calcMDPColPrior(alpha = 16,
                                setList = list(c(1), c(2), c(3), c(4)),
                                setCountMax = 4)

checkAbsDiff(expected = ex2.mdp.expt15,
              answer = ex2.mdp.ans15)

```

Absolute difference: 9.7145e-17

Test passed: TRUE

```

checkAbsRelativeDiff(expected = ex2.mdp.expt15,
                     answer = ex2.mdp.ans15,
                     rel = min(abs(c(ex2.mdp.expt15, ex2.mdp.ans15))))

```

Absolute relative difference: 1.4708e-15

Test passed: TRUE

Test calcLogSubTypeLik Function

```

ex2.subtype.lik.expt = ex2.mdp.expt1*exp(ex2.expt1) +
  ex2.mdp.expt2*exp(ex2.expt2) +
  ex2.mdp.expt3*exp(ex2.expt3) +
  ex2.mdp.expt4*exp(ex2.expt4) +
  ex2.mdp.expt5*exp(ex2.expt5) +
  ex2.mdp.expt6*exp(ex2.expt6) +
  ex2.mdp.expt7*exp(ex2.expt7) +
  ex2.mdp.expt8*exp(ex2.expt8) +
  ex2.mdp.expt9*exp(ex2.expt9) +
  ex2.mdp.expt10*exp(ex2.expt10) +
  ex2.mdp.expt11*exp(ex2.expt11) +
  ex2.mdp.expt12*exp(ex2.expt12) +

```

```

ex2.mdp.expt13*exp(ex2.expt13) +
ex2.mdp.expt14*exp(ex2.expt14) +
ex2.mdp.expt15*exp(ex2.expt15)

ex2.subtype.log.lik.expt = log(ex2.subtype.lik.expt)

elts4AllPartSetMDPPrior = unlist(lapply(elts4AllPartSet,
                                       calcMDPColPrior,
                                       alpha = 16,
                                       setCountMax = 4))
ex2.subtype.log.lik.ans = calcLogSubTypeLik(eltsAllPartSetList = list(elts4AllPartSet),
                                           eltsAllPartSetPriorList = list(elts4AllPartSetMDPPrior),
                                           sample = slvEx2.df,
                                           mkrList = list(colnames(slvEx2.df)),
                                           alphaC = 3,
                                           betaC = 2)

checkAbsDiff(expected = ex2.subtype.log.lik.expt,
              answer = ex2.subtype.log.lik.ans)

## Absolute difference: 3.5527e-15
## Test passed: TRUE

checkAbsRelativeDiff(expected = ex1.subtype.log.lik.expt,
                     answer = ex1.subtype.log.lik.ans,
                     rel = min(abs(c(ex1.subtype.log.lik.expt,
                                     ex1.subtype.log.lik.ans))))

## Absolute relative difference: 3.1794e-16
## Test passed: TRUE

```

Test calcLogSubTypeLik Function

```

ex.subtype.log.lik.expt =
  ex1.subtype.log.lik.expt + ex2.subtype.log.lik.expt
ex.subtype.log.lik.ans = calcLogSubTypeLik(eltsAllPartSetList = list(elts3AllPartSet,
                                                                    elts4AllPartSet),
                                           eltsAllPartSetPriorList = list(elts3AllPartSetMDPPrior,
                                                                    elts4AllPartSetMDPPrior),
                                           sample = cbind(slvEx1.df, slvEx2.df),
                                           mkrList = list(colnames(slvEx1.df), colnames(slvEx2.df)),
                                           alphaC = c(2, 3),
                                           betaC = c(3, 2))

checkAbsDiff(expected = ex.subtype.log.lik.expt,
              answer = ex.subtype.log.lik.ans)

## Absolute difference: 7.1054e-15
## Test passed: TRUE

checkAbsRelativeDiff(expected = ex.subtype.log.lik.expt,
                     answer = ex.subtype.log.lik.ans,

```

```
rel = min(abs(c(ex.subtype.log.lik.expt,  
                ex.subtype.log.lik.ans))))
```

```
## Absolute relative difference: 1.4299e-16
```

```
## Test passed: TRUE
```