March 07 tensor clustering

Yuchen Zeng

March 2019

1 The bottleneck of the sparse.choosekrl()

Listed the trace in the appendix A, what we can see is that the long running time of this function comes from the complex iterations. Therefore, what we can do to speed up this function is:

- using Rcpp to improve the speed of each iteration;
- change the for loop into apply().

2 Two problems with chooseLambda()

2.1 Problem 1: the range of lambda

Sometimes I found it is pretty hard to find a suitable range for lambda. Instead, we have to try many times to find a suitable range for us to find the best lambda. Therefore, we may delete the parameter to input the range of lambda into this function, instead, we can use optimize() to find the best lambda which is different with sparse.choosekrl(). (I already tried this method, and it would super slow:(the result looks better but it may depends on the initial value we take.) The new result was listed in the appendices.

2.2 Problem 2: the penalty of non-zeros parameters

I tries many cases, I found that the lambda with the lowest BIC is always 0. I don't know whether it is because of the range I select is not good or the penalty is too small.

3 New problem about the definition of plot_tensor()

Now we already can get the same plot if the input k,r,l are totally the same with the true k,r,l by using a good definition of quantile. But what if the input k,r,l is different with the true k,r,l. Will the ouput color be totally different from the color of true mus?

4 A reminder for me to change the way to calculate the CER in sparse clustering

5 The results of clustering dnations.mat

The code has been put into appendices. My result is listed as follows. First I tried sparse.choosekrl(), the range is 2:10, 2:10, 2:10, and the result is 5,5,5. The result below is obtained by using k,r,l=5. To avoid the estimated k,r,l is bounded by 5, then I enlarged the range, but then the program is too slow. And I still have not got the updated k,r,l.

5.1 Clusters of countries

```
cluster  1 :
[1] "Egypt"  "India"  "Israel"
cluster  2 :
[1] "UK"  "USA"
cluster  3 :
[1] "Burma"     "Indonesia" "Jordan"
cluster  4 :
[1] "Brazil"      "Netherlands"
cluster  5 :
[1] "China"  "Cuba"   "Poland" "USSR"
```

5.2 Clusters of relationship

```
cluster  1 :
 [1] "economicaid"        "releconomicaid"     "officialvisits"     "warning"
          "violentactions"
 [6] "militaryactions"    "duration"           "negativebehavior"   "severdiplomatic
    "    "expeldiplomats"
[11] "boycottembargo"     "aidenemy"           "negativecomm"       "accusation"
        "protests"
[16] "unoffialacts"       "attackembassy"      "nonviolentbehavior" "timesincewar"
      "lostterritory"
[21] "dependent"          "commonbloc0"
cluster  2 :
[1] "treaties"            "conferences"        "weightedunvote"     "unweightedunvote
   "   "intergovorgs"
[6] "ngo"                 "timesinceally"      "independence"       "
   blockpositionindex"
cluster  3 :
[1] "relintergovorgs" "relngo"          "intergovorgs3"   "ngoorgs3"
cluster  4 :
[1] "embassy"       "reldiplomacy" "commonbloc1"
cluster  5 :
 [1] "reltreaties"         "exportbooks"        "relexportbooks"      "
    booktranslations"
 [5] "relbooktranslations" "tourism"            "reltourism"          "tourism3"
 [9] "emigrants"           "relemigrants"       "emigrants3"          "students"
[13] "relstudents"         "exports"            "relexports"          "exports3"
[17] "militaryalliance"    "commonbloc2"
```

A Rprofile of sparse.choosekrl()

```
$by.self
                   self.time self.pct total.time total.pct
tensorsparse.R#196    393.28    43.17     762.68     83.72
tensorsparse.R#87     269.92    29.63     269.92     29.63
tensorsparse.R#90      72.06     7.91      72.06      7.91
tensorsparse.R#186     54.16     5.94      54.16      5.94
tensorsparse.R#82      32.94     3.62      32.94      3.62
tensorsparse.R#148     27.46     3.01      27.46      3.01
tensorsparse.R#89      25.20     2.77      25.20      2.77
tensorsparse.R#99      19.26     2.11      19.26      2.11
tensorsparse.R#128      6.08     0.67       6.08      0.67
```

```
tensorsparse.R#86        2.22      0.24        2.22       0.24
tensorsparse.R#79        1.78      0.20        1.78       0.20
tensorsparse.R#160       1.38      0.15        1.38       0.15
tensorsparse.R#80        1.28      0.14        1.28       0.14
tensorsparse.R#225       0.82      0.09        0.82       0.09
tensorsparse.R#151       0.80      0.09        0.80       0.09
tensorsparse.R#121       0.44      0.05        0.44       0.05
tensorsparse.R#123       0.36      0.04        0.36       0.04
tensorsparse.R#122       0.34      0.04        0.36       0.04
tensorsparse.R#142       0.30      0.03        0.30       0.03
tensorsparse.R#145       0.10      0.01        0.10       0.01
tensorsparse.R#143       0.08      0.01       34.80       3.82
tensorsparse.R#206       0.08      0.01       23.36       2.56
tensorsparse.R#211       0.08      0.01       22.06       2.42
simulation.R#46          0.06      0.01      911.02     100.00
tensorsparse.R#201       0.06      0.01       19.58       2.15
tensorsparse.R#129       0.06      0.01        0.06       0.01
tensorsparse.R#204       0.04      0.00        2.56       0.28
tensorsparse.R#202       0.04      0.00        2.02       0.22
tensorsparse.R#138       0.04      0.00        0.04       0.00
tensorsparse.R#293       0.04      0.00        0.04       0.00
tensorsparse.R#303       0.04      0.00        0.04       0.00
tensorsparse.R#235       0.02      0.00      910.84      99.98
tensorsparse.R#212       0.02      0.00        5.82       0.64
tensorsparse.R#215       0.02      0.00        4.52       0.50
tensorsparse.R#207       0.02      0.00        2.76       0.30
tensorsparse.R#298       0.02      0.00        0.04       0.00
tensorsparse.R#124       0.02      0.00        0.02       0.00
tensorsparse.R#125       0.02      0.00        0.02       0.00
tensorsparse.R#130       0.02      0.00        0.02       0.00
tensorsparse.R#140       0.02      0.00        0.02       0.00
tensorsparse.R#159       0.02      0.00        0.02       0.00
tensorsparse.R#162       0.02      0.00        0.02       0.00

$by.total
                   total.time total.pct self.time self.pct
simulation.R#46        911.02    100.00      0.06     0.01
tensorsparse.R#235     910.84     99.98      0.02     0.00
tensorsparse.R#316     910.84     99.98      0.00     0.00
tensorsparse.R#196     762.68     83.72    393.28    43.17
tensorsparse.R#87      269.92     29.63    269.92    29.63
tensorsparse.R#90       72.06      7.91     72.06     7.91
tensorsparse.R#186      54.16      5.94     54.16     5.94
tensorsparse.R#143      34.80      3.82      0.08     0.01
tensorsparse.R#82       32.94      3.62     32.94     3.62
tensorsparse.R#148      27.46      3.01     27.46     3.01
tensorsparse.R#89       25.20      2.77     25.20     2.77
tensorsparse.R#206      23.36      2.56      0.08     0.01
tensorsparse.R#211      22.06      2.42      0.08     0.01
tensorsparse.R#201      19.58      2.15      0.06     0.01
tensorsparse.R#99       19.26      2.11     19.26     2.11
tensorsparse.R#128       6.08      0.67      6.08     0.67
tensorsparse.R#212       5.82      0.64      0.02     0.00
tensorsparse.R#215       4.52      0.50      0.02     0.00
tensorsparse.R#210       2.90      0.32      0.00     0.00
tensorsparse.R#207       2.76      0.30      0.02     0.00
tensorsparse.R#204       2.56      0.28      0.04     0.00
tensorsparse.R#86        2.22      0.24      2.22     0.24
tensorsparse.R#209       2.10      0.23      0.00     0.00
tensorsparse.R#202       2.02      0.22      0.04     0.00
tensorsparse.R#79        1.78      0.20      1.78     0.20
```

```
tensorsparse.R#214        1.52       0.17       0.00       0.00
tensorsparse.R#160        1.38       0.15       1.38       0.15
tensorsparse.R#205        1.34       0.15       0.00       0.00
tensorsparse.R#80         1.28       0.14       1.28       0.14
tensorsparse.R#193        1.20       0.13       0.00       0.00
tensorsparse.R#225        0.82       0.09       0.82       0.09
tensorsparse.R#203        0.82       0.09       0.00       0.00
tensorsparse.R#151        0.80       0.09       0.80       0.09
tensorsparse.R#208        0.48       0.05       0.00       0.00
tensorsparse.R#121        0.44       0.05       0.44       0.05
tensorsparse.R#123        0.36       0.04       0.36       0.04
tensorsparse.R#122        0.36       0.04       0.34       0.04
tensorsparse.R#142        0.30       0.03       0.30       0.03
tensorsparse.R#213        0.12       0.01       0.00       0.00
tensorsparse.R#145        0.10       0.01       0.10       0.01
tensorsparse.R#129        0.06       0.01       0.06       0.01
tensorsparse.R#138        0.04       0.00       0.04       0.00
tensorsparse.R#293        0.04       0.00       0.04       0.00
tensorsparse.R#303        0.04       0.00       0.04       0.00
tensorsparse.R#298        0.04       0.00       0.02       0.00
tensorsparse.R#124        0.02       0.00       0.02       0.00
tensorsparse.R#125        0.02       0.00       0.02       0.00
tensorsparse.R#130        0.02       0.00       0.02       0.00
tensorsparse.R#140        0.02       0.00       0.02       0.00
tensorsparse.R#159        0.02       0.00       0.02       0.00
tensorsparse.R#162        0.02       0.00       0.02       0.00

$by.line
                   self.time self.pct total.time total.pct
simulation.R#46         0.06     0.01     911.02    100.00
tensorsparse.R#79       1.78     0.20       1.78      0.20
tensorsparse.R#80       1.28     0.14       1.28      0.14
tensorsparse.R#82      32.94     3.62      32.94      3.62
tensorsparse.R#86       2.22     0.24       2.22      0.24
tensorsparse.R#87     269.92    29.63     269.92     29.63
tensorsparse.R#89      25.20     2.77      25.20      2.77
tensorsparse.R#90      72.06     7.91      72.06      7.91
tensorsparse.R#99      19.26     2.11      19.26      2.11
tensorsparse.R#121      0.44     0.05       0.44      0.05
tensorsparse.R#122      0.34     0.04       0.36      0.04
tensorsparse.R#123      0.36     0.04       0.36      0.04
tensorsparse.R#124      0.02     0.00       0.02      0.00
tensorsparse.R#125      0.02     0.00       0.02      0.00
tensorsparse.R#128      6.08     0.67       6.08      0.67
tensorsparse.R#129      0.06     0.01       0.06      0.01
tensorsparse.R#130      0.02     0.00       0.02      0.00
tensorsparse.R#138      0.04     0.00       0.04      0.00
tensorsparse.R#140      0.02     0.00       0.02      0.00
tensorsparse.R#142      0.30     0.03       0.30      0.03
tensorsparse.R#143      0.08     0.01      34.80      3.82
tensorsparse.R#145      0.10     0.01       0.10      0.01
tensorsparse.R#148     27.46     3.01      27.46      3.01
tensorsparse.R#151      0.80     0.09       0.80      0.09
tensorsparse.R#159      0.02     0.00       0.02      0.00
tensorsparse.R#160      1.38     0.15       1.38      0.15
tensorsparse.R#162      0.02     0.00       0.02      0.00
tensorsparse.R#186     54.16     5.94      54.16      5.94
tensorsparse.R#193      0.00     0.00       1.20      0.13
tensorsparse.R#196    393.28    43.17     762.68     83.72
tensorsparse.R#201      0.06     0.01      19.58      2.15
tensorsparse.R#202      0.04     0.00       2.02      0.22
```

```
tensorsparse.R#203      0.00     0.00      0.82     0.09
tensorsparse.R#204      0.04     0.00      2.56     0.28
tensorsparse.R#205      0.00     0.00      1.34     0.15
tensorsparse.R#206      0.08     0.01     23.36     2.56
tensorsparse.R#207      0.02     0.00      2.76     0.30
tensorsparse.R#208      0.00     0.00      0.48     0.05
tensorsparse.R#209      0.00     0.00      2.10     0.23
tensorsparse.R#210      0.00     0.00      2.90     0.32
tensorsparse.R#211      0.08     0.01     22.06     2.42
tensorsparse.R#212      0.02     0.00      5.82     0.64
tensorsparse.R#213      0.00     0.00      0.12     0.01
tensorsparse.R#214      0.00     0.00      1.52     0.17
tensorsparse.R#215      0.02     0.00      4.52     0.50
tensorsparse.R#225      0.82     0.09      0.82     0.09
tensorsparse.R#235      0.02     0.00    910.84    99.98
tensorsparse.R#293      0.04     0.00      0.04     0.00
tensorsparse.R#298      0.02     0.00      0.04     0.00
tensorsparse.R#303      0.04     0.00      0.04     0.00
tensorsparse.R#316      0.00     0.00    910.84    99.98

$sample.interval
[1] 0.02

$sampling.time
[1] 911.02
```

B The R code to cluster dnations.mat

```
if(!require("rmatio")){
  install.packages("rmatio")
  stopifnot(require("rmatio"))
}
source('tensorsparse.R')

dnations = read.mat("dnations.mat")
country = unlist(dnations$countrynames)
relationship = unlist(dnations$relnnames)
att = unlist(dnations$attnames)
dnation_arr = (2*dnations$R-1)*10
dnation_arr[is.nan(dnation_arr)] = 0


krl = sparse.choosekrl(dnation_arr,2:5,2:5,2:5)
#our result in sparse.choosekrl() is 5,5,5
relationship_label = label2(dnation_arr,5,5,5)
relationship_label$Cs
relationship_label$Ds
relationship_label$Es
for (i in 1:5){
  cat("cluster␣",i,":\n")
  print(country[relationship_label$Cs==i])
}
for (i in 1:5){
  cat("cluster,␣",i,":\n")
  print(relationship[relationship_label$Es==i])
}


$estimated_kr
     [,1] [,2] [,3]
```

```
[1,]    5    5    5

$results.se
, , L = 5

           R = 5    R = 6    R = 7    R = 8    R = 9   R = 10
K = 5  5933.435 5552.789 5296.343 5239.945 5028.949 4997.151
K = 6  5949.399 5561.578 5451.368 5387.178 5076.400 5053.245
K = 7  6063.003 5682.731 5589.186 5381.293 5167.628 5082.386
K = 8  5742.482 5437.832 5518.353 5136.984 4956.522 4992.278
K = 9  5639.228 5157.166 5282.963 5121.966 4776.031 4712.576
K = 10 5594.872 5159.248 5240.639 4707.893 4612.308 4206.844

, , L = 6

           R = 5    R = 6    R = 7    R = 8    R = 9   R = 10
K = 5  5828.189 5473.479 5334.921 5227.326 4973.818 5044.779
K = 6  5726.943 5647.883 5462.790 5387.830 5010.194 5115.050
K = 7  5768.731 5563.020 5581.316 5350.648 5050.824 5140.676
K = 8  5723.458 5391.735 5623.999 5232.224 5044.535 5169.820
K = 9  5558.866 5062.512 5344.152 5208.000 4892.373 4911.731
K = 10 5457.989 5073.254 5266.877 4836.138 4866.966 4547.622

, , L = 7

           R = 5    R = 6    R = 7    R = 8    R = 9   R = 10
K = 5  5810.854 5471.269 5283.968 5144.203 4866.545 4943.979
K = 6  5838.357 5466.073 5281.825 5094.621 4808.000 5018.790
K = 7  5762.995 5426.509 5622.587 5506.223 5209.475 5385.598
K = 8  5463.880 5111.444 5420.685 5347.818 5184.629 5295.328
K = 9  5390.141 4865.092 5120.806 4919.150 4594.657 4629.506
K = 10 5283.692 4900.644 5082.770 4968.200 4621.289 4653.673

, , L = 8

           R = 5    R = 6    R = 7    R = 8    R = 9   R = 10
K = 5  5512.850 5226.016 5170.784 5256.532 4896.963 4856.711
K = 6  5662.549 5408.524 5120.544 5292.055 4805.103 4929.998
K = 7  5482.461 5162.246 5139.782 5507.892 5378.398 5293.933
K = 8  4996.948 4832.955 4891.994 5094.060 5166.374 5103.209
K = 9  5029.056 4751.835 4652.263 4683.257 4355.256 4188.731
K = 10 5184.933 4761.462 4847.815 4864.643 4751.329 4585.901

, , L = 9

           R = 5    R = 6    R = 7    R = 8    R = 9   R = 10
K = 5  5092.393 4916.785 5068.483 5194.838 5047.737 4848.045
K = 6  5215.378 5059.897 4890.586 5100.640 4822.050 4816.145
K = 7  5117.740 4956.342 5195.852 5595.416 5425.984 5175.471
K = 8  4584.519 4487.400 4849.573 5023.491 5179.372 4931.752
K = 9  4602.946 4211.446 4389.301 4535.432 4390.707 4058.499
K = 10 4720.300 4236.758 4668.369 4770.880 4782.131 4351.787

, , L = 10

           R = 5    R = 6    R = 7    R = 8    R = 9   R = 10
K = 5  5380.532 5204.565 5007.741 4794.561 4751.250 4765.771
K = 6  5721.548 5310.215 5074.755 4969.275 4523.610 4698.753
K = 7  5430.110 5045.681 4870.134 4780.291 4709.825 4785.550
K = 8  5057.639 4789.875 4722.049 4512.756 4590.281 4724.512
K = 9  5211.760 4679.333 4779.904 4492.020 4322.838 4312.883
```

```
K = 10 5036.333 4629.174 4550.536 4450.234 4173.690 4148.455


$results.mean
, , L = 5

          R = 5    R = 6    R = 7    R = 8    R = 9   R = 10
K = 5  91316.56 90666.84 91007.17 90506.33 90560.17 90811.23
K = 6  91134.08 90157.61 90300.61 89968.64 89965.23 90055.84
K = 7  91426.30 90391.26 90823.57 90444.13 90089.88 90132.16
K = 8  91995.75 90910.09 90781.99 90597.28 90261.71 90455.54
K = 9  91285.82 90386.53 90315.76 90302.44 90454.84 90276.06
K = 10 91622.14 90551.01 90761.86 90521.31 90130.42 89732.69

, , L = 6

          R = 5    R = 6    R = 7    R = 8    R = 9   R = 10
K = 5  89246.44 88640.09 88880.74 88491.18 88263.54 88412.17
K = 6  88960.24 88061.17 88306.74 87972.72 87629.68 87729.48
K = 7  89433.58 88451.49 88968.78 88529.74 87617.66 87653.90
K = 8  89867.22 88806.69 88765.84 88345.07 87746.58 88087.84
K = 9  89244.91 88252.22 88331.16 88358.05 88107.40 88164.43
K = 10 89459.62 88283.11 88663.97 88378.67 87758.58 87521.30

, , L = 7

          R = 5    R = 6    R = 7    R = 8    R = 9   R = 10
K = 5  87469.56 86940.52 87227.04 86891.31 86754.25 86850.31
K = 6  87368.42 86452.15 86713.71 86527.74 86193.92 86104.30
K = 7  87970.76 86880.88 87435.06 87279.84 86447.41 86419.24
K = 8  88241.20 87180.46 87193.66 87161.24 86680.75 86884.03
K = 9  87334.72 86449.21 86632.21 86708.73 86750.90 86680.85
K = 10 87779.51 86519.68 86997.72 87151.32 86465.66 86276.95

, , L = 8

          R = 5    R = 6    R = 7    R = 8    R = 9   R = 10
K = 5  86422.09 86206.14 86438.25 86110.52 85629.39 85552.58
K = 6  86332.71 85416.49 85792.98 85573.32 84913.31 84676.26
K = 7  86880.36 85707.67 86238.97 85915.00 84863.54 84813.55
K = 8  87200.41 86089.49 85923.00 85966.07 85054.55 85194.36
K = 9  85940.44 85111.48 85133.54 85283.64 85116.78 84996.34
K = 10 86105.45 85222.57 85291.23 85723.66 84669.57 84415.23

, , L = 9

          R = 5    R = 6    R = 7    R = 8    R = 9   R = 10
K = 5  85841.37 85611.99 85852.10 85399.25 84934.12 85056.68
K = 6  85716.25 84737.80 84945.76 84584.84 84001.23 84012.00
K = 7  86363.89 85110.12 85590.97 85198.99 84043.05 84297.62
K = 8  86718.44 85476.28 85227.70 85372.00 84439.88 84870.34
K = 9  85325.82 84318.67 84585.79 84677.73 84404.44 84471.26
K = 10 85483.51 84398.69 84772.31 85061.83 84019.74 83971.74

, , L = 10

          R = 5    R = 6    R = 7    R = 8    R = 9   R = 10
K = 5  84106.95 83617.89 83885.61 83891.03 83000.10 83113.09
K = 6  83734.25 82784.06 83200.32 83144.59 82530.81 82586.39
K = 7  84381.33 83166.93 83860.84 83857.90 82676.44 82943.25
K = 8  84729.02 83675.82 83679.58 83957.34 83079.35 83646.69
```

```
K = 9  83415.21 82587.72 82719.32 83231.43 82858.03 82939.46
K = 10 83952.36 82996.90 83421.94 83931.18 82970.53 82946.14
```

C The simulation result of chooseLambda2()

The result I got by using chooseLambda() is always 0 (may because the range I selected is always bad). And here is the result I got from chooseLambda2():

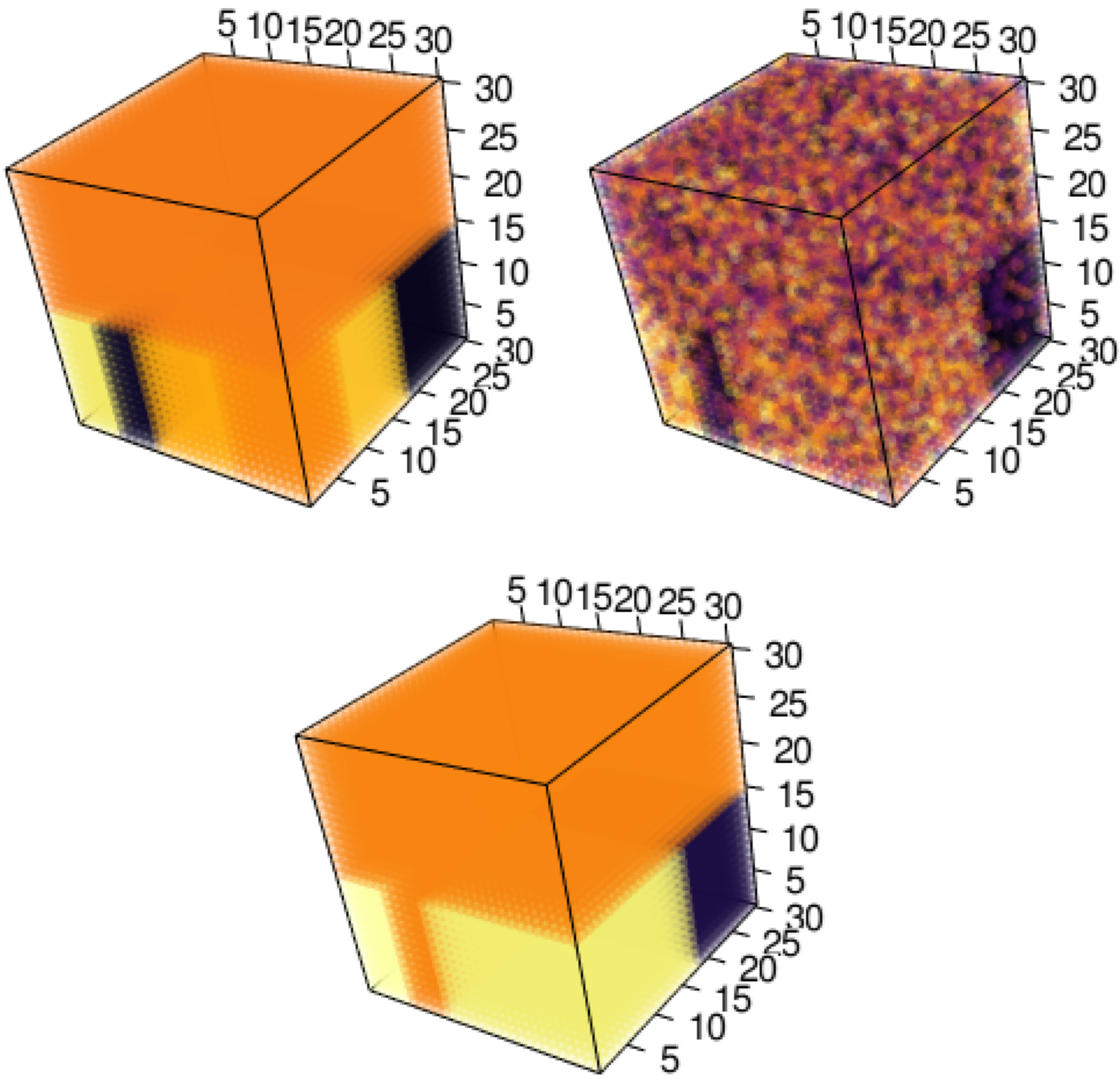

Figure 1: truth,input,output