Overview

I will discuss the performance of a rejection sampler for small complex Bingham matrices. In particular, I confirm that a statistic of the samples matches a known property of the distribution, and show how the acceptance rate varies. Overall, I believe the function produces *correct samples* for each of the parameters tested. However, the *acceptance rate can be quite different* depending on the parameters.

Rejection sampler

We wish to generate samples $U \sim CB(G, H)$, where G is $p \times p$ Hermitian positive definite, and H is $p \times p$ diagonal with real entries. The method is based on that described in supplemental materials to Hoff's 2009 Simulation paper. Very roughly:

- generate $W \sim CW(\nu, S)$
    - where $S = (\delta I_p - G)^{-1}$
    - δ is chosen to make S positive definite
    - and ν can be chosen
- let $W = ULU^H$
- determine whether to accept U as a sample
    - density of W can be written as $p(W) = p(U, L) = p(U|L)p(L)$
    - $U|L$ has a $CB(G, L)$ distribution
    - leads to a convenient expression for acceptance ratio, r

In practice, I have implemented this sampler in (almost) exactly the same way as the `rbing.Op` function in Hoff's `rstiefel` package.

How to compare samples to properties of known distribution

We wish to generate many samples from a known CB distribution, and determine if the samples are indeed from that distribution. I have not been able to find many useful results for known properties of the (complex) matrix Bingham distribution, so I do not have convenient known quantities to test like the mean, distributions of eigenvalues, etc.

In Hoff's 2009 Eigenmodel paper, he notes that, when the entries of A and H are distinct, the $CGB(A, H, V)$ distribution has modes at V and $\{VS\}$, where S is any matrix that flips the sign of one or more columns of V. The $CGB(A, H, V)$ distribution is exactly the $CB(G, H) = CB(VAV^H, H)$ distribution that we want to sample.

I propose comparing the "average" columns sampled to the parameter V. However, since the distribution is antipodally symmetric (the columns represent axes), taking the simple average of the columns averages out to 0. Instead, I will adapt an idea from Hoff 2009 for a posterior point estimate of a matrix parameter which has a Bingham full conditional distribution. In Section 4, he proposes "a posterior point estimate of V can be obtained from the eigenvector matrix of the posterior mean of VAV^T, obtained by averaging across samples of the Markov chain".

Thus, for a sample U_i, I propose calculating $\Sigma_i = U_i C U_i^H$ for an arbitrary diagonal matrix C with positive real diagonal entries. The matrix C is included since otherwise, $U_i U_i^H = I_p$ because U_i is unitary. Then, find the sample average $\widehat{\Sigma}$, and compare the eigenvectors of $\widehat{\Sigma}$ to V.

Simulations of different distributions

Goal:

- Choose fixed G, H parameters for CB(G, H)
- Generate multiple samples $U \sim CB(G, H)$
- Confirm a summary of the sample matches a known characteristic of the distribution

Choose the G and H parameters in the following way. Let $G = VAV^H$, and choose some fixed V. Then, we will vary the eigenvalues in A and the entries of H. Specifically, for each we will consider values that are either (small or large) and (close or far). For example, values 4 and 1 are small and close, while 10040 and 10001 are large and far. There are thus four settings each for G and H, making 16 total combinations to compare.

For the examples and simulations, I let $V = 2^{-1/2} \begin{pmatrix} 1 & -1 \\ 1 & 1 \end{pmatrix}$.

Overall, I believe the function produces *correct samples* for each of the parameters tested. However, the *acceptance rate can be quite different* – if the A or H entries are far apart, more rejections are needed, on average.

One example

Below, we show the performance of letting $diag(A) = diag(H) = (4, 1)$. 250 samples are generated from the CB(G, H) distribution.

```
## [1] "compare the estimate with some different C matrices, to confirm they are the same"
```

```
##                     [,1]                  [,2]
## [1,] -0.7211379+0.00000000i -0.6927915+0.00000000i
## [2,] -0.6925058-0.01989393i  0.7208406+0.02070792i
```

```
##                     [,1]                  [,2]
## [1,] -0.7211379+0.00000000i -0.6927915+0.00000000i
## [2,] -0.6925058-0.01989393i  0.7208406+0.02070792i
```

```
##                     [,1]                  [,2]
## [1,] -0.7211379+0.00000000i -0.6927915+0.00000000i
## [2,] -0.6925058-0.01989393i  0.7208406+0.02070792i
```

```
## [1] "compare to the known eigenvectors of G"
```

```
##           [,1]       [,2]
## [1,] 0.7071068 -0.7071068
## [2,] 0.7071068  0.7071068
```

```
## [1] "compute distance between estimated columns and true columns."
```

```
## [1] "Close to 2 or 0 means the estimated column is close to the true axis."
```

```
## [1] 1.99979853 0.02867686
```

```
## [1] "show a summary of how many rejections occurred for each sample"
```

```
##    Min. 1st Qu.  Median   Mean 3rd Qu.   Max.
##       0       0       0      1       1      7
```

```
## 95% 99%
## 4.00 6.51
```

Another example

Next, an example is shown with large and far G eigenvalues, and the same H values as the previous example (small and close). That is, $diag(A) = (10040, 10001)$ and $diag(H) = (4, 1)$.

```
## [1] "Estimate of columns"
```

```
##                           [,1]                      [,2]
## [1,] -0.7097746+0.000000000i -0.7044289+0.000000000i
## [2,] -0.7044255+0.002200499i  0.7097711-0.002217198i
```

```
## [1] "compare to the known eigenvectors of G"
```

```
##            [,1]        [,2]
## [1,] 0.7071068 -0.7071068
## [2,] 0.7071068  0.7071068
```

```
## [1] "compute distance between estimated columns and true columns."
```

```
## [1] "Close to 2 or 0 means the estimated column is close to the true axis."
```

```
## [1] 1.999995213 0.004380147
```

```
## [1] "show a summary of how many rejections occurred for each sample"
```

```
##    Min. 1st Qu.  Median   Mean 3rd Qu.    Max.
##    0.00    5.00   12.00  18.63   27.00  103.00
```

```
##   95%   99%
## 61.10 86.02
```

Further simulations

This method of sampling was repeated for all 16 combinations of A and H entries. In the simulations below, 500 samples were drawn from each distribution. In all cases, the estimated columns are close to the true eigenvectors of G (since the distance between the estimated columns and the true axes are all close to 2 or 0, as shown below).

```
##             [,1]          [,2]
##   [1,] 1.999959 0.0128841729
##   [2,] 1.999999 0.0020189788
##   [3,] 1.999973 0.0103462887
##   [4,] 1.999965 0.0117262121
##   [5,] 1.999993 0.0054495985
##   [6,] 2.000000 0.0003278992
##   [7,] 1.999998 0.0029238687
##   [8,] 2.000000 0.0007504832
##   [9,] 1.999907 0.0194778329
## [10,] 1.999994 0.0050061881
## [11,] 1.999961 0.0124294092
## [12,] 2.000000 0.0011531523
## [13,] 1.999997 0.0037092593
## [14,] 2.000000 0.0011412371
## [15,] 1.999998 0.0028074121
## [16,] 1.999999 0.0015292226
```

The table below summarizes the number of rejections needed for the different parameter combinations. In general, if the eigenvalues of G and entries of H are close, then the sampler is fairly efficient (acceptance rate $\approx 50\%$). If one or both sets of values are far apart, then the acceptance rate becomes quite poor (about $1/20$ or $1/250$).

Table 1: Percentiles and mean numbers of rejections for different eigenvalue combinations

G1	G2	H1	H2	0%	25%	50%	75%	95%	99%	100%	means
4	1	4	1	0	0.00	1.0	2.00	4.00	7.00	10	1.052
4	1	40	1	0	6.00	14.0	27.00	64.00	87.01	101	19.656
4	1	10004	10001	0	0.00	0.5	2.00	4.00	6.00	10	1.026
4	1	10040	10001	0	5.00	13.0	24.00	58.00	81.03	92	18.106
40	1	4	1	0	5.00	13.0	29.00	59.00	88.10	117	19.852
40	1	40	1	0	73.00	179.0	358.25	774.05	1087.89	1463	251.138
40	1	10004	10001	0	6.00	14.0	28.00	60.05	101.00	154	20.234
40	1	10040	10001	0	79.00	174.5	337.00	798.20	1210.61	1503	255.874
10004	10001	4	1	0	0.00	0.0	1.00	4.00	6.00	10	1.030
10004	10001	40	1	0	6.00	13.0	27.00	59.05	83.00	169	18.928
10004	10001	10004	10001	0	0.00	0.0	2.00	4.00	5.01	7	1.032
10004	10001	10040	10001	0	6.00	13.5	30.00	65.00	83.04	137	20.406
10040	10001	4	1	0	5.00	13.0	25.00	53.05	94.09	149	18.354
10040	10001	40	1	0	63.00	175.0	357.00	809.85	1153.01	1519	257.788
10040	10001	10004	10001	0	4.75	12.0	23.00	60.05	87.00	139	17.584
10040	10001	10040	10001	2	83.50	198.0	369.25	732.60	1119.04	1754	266.396

An example with complex eigenvectors

The previous example had real parameters to the CB distribution. In this example, we use $G = VAV^H$ with complex eigenvectors making up V. This will help to test whether the distribution is off by, say, a complex conjugate somewhere. I let $A = H = diag(4, 1)$. The eigenvectors V are randomly generated by taking the eigenvectors from a complex Wishart matrix. 1000 samples are generated from the $CB(VAV^H, H)$ distribution. I compare the average estimated eigenvectors to V and the distances of those columns from the true axes represented by V, and check the number of rejections.

The estimated eigenvectors are close to the columns of V, and the number of rejections is similar to the first example in this section (acceptance rate $\approx 50\%$ with close eigenvalue/diagonal entries).

```
## [1] "The eigendecomposition of G"


## eigen() decomposition
## $values
## [1] 4 1
##
## $vectors
##                      [,1]                 [,2]
## [1,]   0.1792917+0.0000000i -0.983796+0.0000000i
## [2,] -0.8421477-0.5085683i -0.153477-0.0926839i


## [1] "Estimate of columns"


##                      [,1]                  [,2]
## [1,]   0.1774487+0.0000000i -0.9841301+0.00000000i
## [2,] -0.8447014-0.5049669i -0.1523083-0.09105067i


## [1] "compare to the known eigenvectors of G"


##                      [,1]                 [,2]
## [1,]   0.1792917+0.0000000i -0.983796+0.0000000i
## [2,] -0.8421477-0.5085683i -0.153477-0.0926839i


## [1] "compute distance between estimated columns and true columns."


## [1] "Close to 2 or 0 means the estimated column is close to the true axis."


## [1] 0.004784151 0.002035938


## [1] "show a summary of how many rejections occurred for each sample"


##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   0.000   0.000   0.500   1.026   2.000  11.000


##  95%  99%
## 4.00 6.01
```

Number of rejections for randomized parameters

In the following examples, I investigate the amount of rejections for samples with different parameters, to understand if the previous results were just due to lucky choices. In the first example, the eigenvalues of G and the entries of H are $[7 + Unif(-1, 1), 3 + Unif(-1, 1)]$. This is closest to the 1st combination of settings in Table 1. The eigenvectors of G are the eigenvectors from a $CW(3, I_2)$ random matrix. 10,000 repetitions are performed. The mean number of rejections is 1.844 (compared to 1.052 earlier), and both simulations had a median of 1 rejection. Thus, the performance is somewhat worse than the earlier simulation, although not drastically different.

```
## [1] "Summary of rejections for close G eigenvalues"
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   0.000   0.000   1.000   1.844   3.000  24.000
```

```
## 95% 99%
##    7  11
```

Next, we use eigenvalues for G that are further apart. Now, the eigenvalues of G are $[60+Unif(-10,10), 21+Unif(-10,10)]$. The eigenvectors of G and the entries of H are randomly generated in the same way as the previous example. This is closest to the 5th combination of settings in Table 1. 1000 repetitions were performed (due to this simulation taking longer). The mean number of rejections is 25.23 (compared to 19.852 in the earlier simulations). The median is 17 (compared to 13 earlier). Similar to the previous example, the performance is somewhat worse than the previous simulations, although not drastically different.

```
## [1] "Summary of rejections for far G eigenvalues"
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    0.00    6.00   17.00   25.23   36.00  205.00
```

```
##    95%    99%
##   75.0  125.1
```

Sampling 3x3 matrices

Now, I briefly demonstrate the rejection sampler for 3×3 matrices. The eigenvectors of G are randomly generated from eigenvectors of a $CW(4, I_3)$ matrix. The eigenvalues of G and diagonal entries of H are $(7, 4, 1)$. 100 repetitions are performed. The "average" columns are close to the true axes. However, the number of rejections is high (mean $= 32.22$).

```
## eigen() decomposition
## $values
## [1] 7 4 1
##
## $vectors
##                      [,1]                [,2]                [,3]
## [1,] -0.91176060+0.00000000i -0.2761502+0.0000000i -0.3040291+0.0000000i
## [2,]  0.08478778+0.27754469i -0.6048889-0.2007716i  0.2951494-0.6499746i
## [3,]  0.27558026+0.09234771i -0.3073758-0.6504317i -0.5472548+0.3138445i
```

```
## [1] "compare the estimate with some different C matrices, to confirm they are the same"
```

```
##                      [,1]                [,2]                [,3]
## [1,] -0.91619544+0.00000000i -0.2762504+0.0000000i -0.2902958+0.0000000i
## [2,]  0.07372613+0.26920598i -0.6015951-0.2383520i  0.3398030-0.6228147i
## [3,]  0.27077974+0.09673084i -0.3045942-0.6420122i -0.5647443+0.3056600i
```

```
##                      [,1]                [,2]                [,3]
## [1,] -0.91619544+0.00000000i -0.2762504+0.0000000i -0.2902958+0.0000000i
## [2,]  0.07372613+0.26920598i -0.6015951-0.2383520i  0.3398030-0.6228147i
## [3,]  0.27077974+0.09673084i -0.3045942-0.6420122i -0.5647443+0.3056600i
```

```
##                       [,1]                  [,2]                  [,3]
## [1,] -0.9161946+0.00000000i -0.2762513+0.0000000i -0.2902974+0.0000000i
## [2,]  0.0737264+0.26920664i -0.6015969-0.2383513i  0.3398040-0.6228124i
## [3,]  0.2707810+0.09673287i -0.3045890-0.6420129i -0.5647482+0.3056548i


## [1] "compare to the known eigenvectors of G"


##                        [,1]                 [,2]                  [,3]
## [1,] -0.91176060+0.00000000i 0.2761502+0.0000000i -0.3040291+0.0000000i
## [2,]  0.08478778+0.27754469i 0.6048889+0.2007716i  0.2951494-0.6499746i
## [3,]  0.27558026+0.09234771i 0.3073758+0.6504317i -0.5472548+0.3138445i


## [1] "compute distance between estimated columns and true columns."


## [1] "Close to 2 or 0 means the estimated column is close to the true axis."


## [1] 0.01593169 1.99962452 0.05738536


## [1] "show a summary of how many rejections occurred for each sample"


##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    0.00   11.00   25.00   32.22   43.00  136.00


##     95%     99%
##   88.25  135.01
```

When one of the eigenvalues for G is farther, i.e. (20, 4, 1), the rejection sampler takes even longer (mean number of rejections is 594.9).

```
## eigen() decomposition
## $values
## [1] 20  4  1
##
## $vectors
##                        [,1]                 [,2]                  [,3]
## [1,] -0.91176060+0.00000000i 0.2761502+0.0000000i -0.3040291+0.0000000i
## [2,]  0.08478778+0.27754469i 0.6048889+0.2007716i  0.2951494-0.6499746i
## [3,]  0.27558026+0.09234771i 0.3073758+0.6504317i -0.5472548+0.3138445i


## [1] "compare the estimate with some different C matrices, to confirm they are the same"


##                        [,1]                  [,2]                  [,3]
## [1,] -0.90793502+0.00000000i -0.2775188+0.0000000i -0.3140658+0.0000000i
## [2,]  0.07213641+0.29149880i -0.5972103-0.2464112i  0.3191748-0.6249589i
## [3,]  0.27719822+0.09295128i -0.2744944-0.6559402i -0.5588021+0.3108966i


##                        [,1]                  [,2]                  [,3]
## [1,] -0.90793502+0.00000000i -0.2775188+0.0000000i -0.3140658+0.0000000i
## [2,]  0.07213641+0.29149880i -0.5972103-0.2464112i  0.3191748-0.6249589i
## [3,]  0.27719822+0.09295128i -0.2744944-0.6559402i -0.5588021+0.3108966i
```

```
##                        [,1]                 [,2]                 [,3]
## [1,] -0.90793506+0.00000000i -0.2775190+0.0000000i -0.3140655+0.0000000i
## [2,]  0.07213611+0.29149907i -0.5972094-0.2464131i  0.3191756-0.6249585i
## [3,]  0.27719786+0.09295137i -0.2744938-0.6559404i -0.5588019+0.3108973i


## [1] "compare to the known eigenvectors of G"


##                        [,1]               [,2]                 [,3]
## [1,] -0.91176060+0.00000000i 0.2761502+0.0000000i -0.3040291+0.0000000i
## [2,]  0.08478778+0.27754469i 0.6048889+0.2007716i  0.2951494-0.6499746i
## [3,]  0.27558026+0.09234771i 0.3073758+0.6504317i -0.5472548+0.3138445i


## [1] "compute distance between estimated columns and true columns."


## [1] "Close to 2 or 0 means the estimated column is close to the true axis."


## [1] 0.01929745 1.99918600 0.03802328


## [1] "show a summary of how many rejections occurred for each sample"


##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     2.0   193.8   431.0   594.9   821.5  3359.0


##      95%     99%
## 1701.45 2642.24
```

Computation time

This demonstrates the order of run times for sampling different size matrices with different parameters.

```
## Unit: microseconds
##                      expr      min         lq       mean    median         uq
##  my.rCbing.Op(G2c, H2)  619.767   856.3045   2034.747  1312.180   2544.921
##  my.rCbing.Op(G2f, H2)  661.434  3589.6865  12765.491  8358.979  18502.052
##  my.rCbing.Op(G3c, H3)  854.181  5954.6030  22896.239 15590.297  34229.400
##  my.rCbing.Op(G3f, H3) 1765.351 88085.6660 373686.420 255078.303 454527.187
##         max neval
##    16061.61   100
##    72013.55   100
##   202689.44   100
##  1903698.92   100
```

Comparison to sampler for real Bingham matrices

Finally, in this section, I briefly demonstrate the performance of the same technique for real Bingham matrices, using `rbing.Op` from the `rstiefel` package. The samples have "average" columns that are close to the "true" axes. The number of rejections are quite smaller than those for complex matrices. Curiously, the number of rejections seems to decrease for 2×2 matrices when the parameter eigenvalues are very far apart. The mean number of rejections for a 3×3 matrix with close eigenvalues is about 3.5, much smaller than that for complex matrices.

Real 2x2 matrices with close G eigenvalues

```
## [1] "compare the estimate with some different C matrices, to confirm they are the same"
```

```
##           [,1]        [,2]
## [1,] 0.7016918 -0.7124806
## [2,] 0.7124806  0.7016918
```

```
##           [,1]        [,2]
## [1,] 0.7016918 -0.7124806
## [2,] 0.7124806  0.7016918
```

```
##           [,1]        [,2]
## [1,] 0.7016918 -0.7124806
## [2,] 0.7124806  0.7016918
```

```
## [1] "compare to the known eigenvectors of G"
```

```
##           [,1]        [,2]
## [1,] 0.7071068 -0.7071068
## [2,] 0.7071068  0.7071068
```

```
## [1] "compute distance between estimated columns and true columns."
```

```
## [1] "Close to 2 or 0 means the estimated column is close to the true axis."
```

```
## [1] 0.007628887 0.007628887
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   0.000   0.000   0.000   0.755   1.000   7.000
```

```
## 95% 99%
##   3   5
```

Real 2x2 matrices with far G eigenvalues

```
## [1] "compare the estimate with some different C matrices, to confirm they are the same"
```

```
##           [,1]        [,2]
## [1,] 0.7069423 -0.7072712
## [2,] 0.7072712  0.7069423
```

```
##           [,1]        [,2]
## [1,] 0.7069423 -0.7072712
## [2,] 0.7072712  0.7069423
```

```
##           [,1]        [,2]
## [1,] 0.7069423 -0.7072712
## [2,] 0.7072712  0.7069423
```

```
## [1] "compare to the known eigenvectors of G"

##            [,1]       [,2]
## [1,] 0.7071068 -0.7071068
## [2,] 0.7071068  0.7071068

## [1] "compute distance between estimated columns and true columns."

## [1] "Close to 2 or 0 means the estimated column is close to the true axis."

## [1] 0.0002325479 0.0002325479

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   0.000   0.000   0.000   0.221   0.000   3.000

## 95% 99%
##   1   2
```

Real 3x3 matrices with close G eigenvalues

```
## eigen() decomposition
## $values
## [1] 7 4 1
##
## $vectors
##             [,1]        [,2]       [,3]
## [1,]  0.87247507 -0.09576798  0.4791824
## [2,] -0.01087041 -0.98416868 -0.1769007
## [3,]  0.48853771  0.14913252 -0.8597037

## [1] "Estimate from sampled matrices"

##             [,1]       [,2]       [,3]
## [1,]  0.87404581 -0.1085633  0.4735588
## [2,] -0.02190054 -0.9825278 -0.1848227
## [3,]  0.48534965  0.1511723 -0.8611519

## [1] "compare to the known eigenvectors of G"

##             [,1]        [,2]       [,3]
## [1,]  0.87247507 -0.09576798  0.4791824
## [2,] -0.01087041 -0.98416868 -0.1769007
## [3,]  0.48853771  0.14913252 -0.8597037

## [1] "compute distance between estimated columns and true columns."

## [1] "Close to 2 or 0 means the estimated column is close to the true axis."

## [1] 0.011588559 0.013060406 0.009822477

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   0.000   1.000   2.000   3.488   5.000  28.000

##    95%    99%
## 11.00  15.01
```

Appendix

```
#####
# Lambdak_gibbs_densCovar
#####

# Lambdak_gibbs_densCovar
```