

Rib Fracture Clustering Analysis

Kiersten C Woodyard De Brito

3/23/2023

Coding the Scaled Rib Fracture Grids into R

Using Anterior-Posterior position as vector to list Ribs 1-12. They are then bound together to form a matrix.

```
mvc_R_post = c(2.432, 1.720, 1.661, 1.779, 1.305, 1.364, 1.423, 1.364, 1.127, 0.949,
0.771, 0.712)
mvcR_postlat = c(0.593, 0.474, 0.830, 0.949, 1.068, 0.949, 0.949, 1.246, 1.186,
1.246, 1.008, 0.356)
mvc_R_lat = c(0.534, 0.890, 2.195, 3.144, 3.737, 4.270, 4.093, 2.847, 2.135, 0.771,
0.356, 0.059)
mvc_R_antlat = c(1.008,2.788, 3.499, 4.508, 4.093, 3.796, 3.144, 2.610, 1.127, 0.356,
0.059, 0.000)
mvc_R_ant = c(1.898, 2.550, 2.966, 2.906, 3.321, 3.084, 2.313, 0.949, 0.474, 0.059,
0.000, 0.000)
mvc_L_post = c(1.271,1.222, 1.466, 1.466, 1.662, 1.857, 1.613, 1.320, 1.564, 1.320,
1.271, 0.587)
mvc_L_postlat = c(0.391,0.538, 0.929, 1.173, 1.466, 1.515, 1.613, 1.906, 1.760,
1.564, 0.929, 0.782)
mvc_L_lat = c(0.489,1.320, 2.688, 4.203, 4.839, 4.985, 4.497, 3.666, 2.493, 1.711,
0.489, 0.147)
mvc_L_antlat = c(0.635,2.395, 3.861, 4.008, 4.008, 3.861, 2.835, 1.417, 0.587, 0.147,
0.049, 0.000)
mvc_L_ant = c(1.369, 2.004, 1.613, 1.711, 1.760, 1.906, 1.564, 0.782, 0.587, 0.147,
0.049, 0.000)
mvc_Rmatrix = cbind(mvc_R_post, mvcR_postlat, mvc_R_lat, mvc_R_antlat, mvc_R_ant)
mvc_Lmatrix = cbind(mvc_L_ant, mvc_L_antlat, mvc_L_lat, mvc_L_postlat, mvc_L_post)
rownames(mvc_Rmatrix) = c("Rib 1", "Rib 2", "Rib 3", "Rib 4", "Rib 5", "Rib 6",
"Rib 7", "Rib 8", "Rib 9", "Rib 10", "Rib 11", "Rib 12")
rownames(mvc_Lmatrix) = c("Rib 1", "Rib 2", "Rib 3", "Rib 4", "Rib 5", "Rib 6",
"Rib 7", "Rib 8", "Rib 9", "Rib 10", "Rib 11", "Rib 12")
colnames(mvc_Rmatrix) =c("R Posterior", "R Posterior-Lateral", "R Lateral",
"R Anterior-Lateral", "R Anterior")
colnames(mvc_Lmatrix) =c("L Anterior", "L Anterior-Lateral", " L Lateral",
"L Posterior-Lateral", "L Posterior")

mcc_R_post = c(1.774, 1.935, 1.935, 2.742, 2.419, 2.419, 2.742, 2.742, 1.935, 1.452,
0.484, 0.484)
mcc_R_postlat = c(0.484,0.968, 1.774,2.742, 2.742, 2.581, 2.419, 1.935, 0.968, 0.806,
0.484, 0.161)
mcc_R_lat = c(0.161,1.452, 2.097, 3.387, 3.548, 4.516, 4.194, 3.226, 2.581, 0.645,
0.161, 0.000)
mcc_R_antlat = c(0.484,2.419,4.032, 4.839, 4.032, 3.548, 1.774, 0.806, 0.323, 0.161,
0.000, 0.000)
mcc_R_ant = c(0.968,1.129, 1.290, 1.935, 1.613, 1.452, 1.129, 0.968, 0.000, 0.000,
```

```

0.000, 0.000)
mcc_L_ant = c(1.136,1.136, 1.010, 1.389, 1.389, 1.768, 1.389, 0.758, 0.126, 0.253,
0.126, 0.126)
mcc_L_antlat = c(0.631,1.515, 2.652, 2.273, 2.399, 2.525, 1.641, 1.389, 0.379, 0.000,
0.000, 0.000)
mcc_L_lat = c(0.253, 1.263, 2.525, 3.409, 3.914, 3.662, 3.535, 2.020, 1.515, 0.884,
0.253, 0.000)
mcc_L_postlat = c(0.884,1.515, 2.778,3.535, 4.040, 3.662, 3.409, 3.157, 2.399, 1.515,
1.136, 0.758)
mcc_L_post = c(1.389, 2.020,2.399, 2.020, 2.652, 2.652, 3.157, 2.273, 1.515, 0.884,
0.505, 0.505)
mcc_Rmatrix = cbind(mcc_R_post, mcc_R_postlat, mcc_R_lat, mcc_R_antlat, mcc_R_ant)
mcc_Lmatrix = cbind(mcc_L_ant, mcc_L_antlat, mcc_L_lat, mcc_L_postlat, mcc_L_post)
rownames(mcc_Rmatrix) = c("Rib 1", "Rib 2", "Rib 3", "Rib 4", "Rib 5", "Rib 6", "Rib
7", "Rib 8", "Rib 9", "Rib 10", "Rib 11", "Rib 12")
rownames(mcc_Lmatrix) = c("Rib 1", "Rib 2", "Rib 3", "Rib 4", "Rib 5", "Rib 6", "Rib
7", "Rib 8", "Rib 9", "Rib 10", "Rib 11", "Rib 12")
colnames(mcc_Rmatrix) =c("R Posterior", "R Posterior-Lateral", "R Lateral", "R Anteri
or-Lateral", "R Anterior")
colnames(mcc_Lmatrix) =c("L Anterior", "L Anterior-Lateral", " L Lateral", "L Posteri
or-Lateral", "L Posterior")

```

→ Combine the Left and Right sides for MCC

```

mcc_matrix_comb = cbind(mcc_Rmatrix, mcc_Lmatrix)
mcc_matrix_comb

```

##	R Posterior	R Posterior-Lateral	R Lateral	R Anterior-Lateral	R Anterior
## Rib 1	1.774	0.484	0.161	0.484	0.968
## Rib 2	1.935	0.968	1.452	2.419	1.129
## Rib 3	1.935	1.774	2.097	4.032	1.290
## Rib 4	2.742	2.742	3.387	4.839	1.935
## Rib 5	2.419	2.742	3.548	4.032	1.613
## Rib 6	2.419	2.581	4.516	3.548	1.452
## Rib 7	2.742	2.419	4.194	1.774	1.129
## Rib 8	2.742	1.935	3.226	0.806	0.968
## Rib 9	1.935	0.968	2.581	0.323	0.000
## Rib 10	1.452	0.806	0.645	0.161	0.000
## Rib 11	0.484	0.484	0.161	0.000	0.000
## Rib 12	0.484	0.161	0.000	0.000	0.000
##	L Anterior	L Anterior-Lateral	L Lateral	L Posterior-Lateral	L Posterior
## Rib 1	1.136	0.631	0.253	0.884	1.389
## Rib 2	1.136	1.515	1.263	1.515	2.020
## Rib 3	1.010	2.652	2.525	2.778	2.399
## Rib 4	1.389	2.273	3.409	3.535	2.020
## Rib 5	1.389	2.399	3.914	4.040	2.652
## Rib 6	1.768	2.525	3.662	3.662	2.652
## Rib 7	1.389	1.641	3.535	3.409	3.157
## Rib 8	0.758	1.389	2.020	3.157	2.273
## Rib 9	0.126	0.379	1.515	2.399	1.515
## Rib 10	0.253	0.000	0.884	1.515	0.884
## Rib 11	0.126	0.000	0.253	1.136	0.505
## Rib 12	0.126	0.000	0.000	0.758	0.505

→ Combine the Left and Right sides for MVC

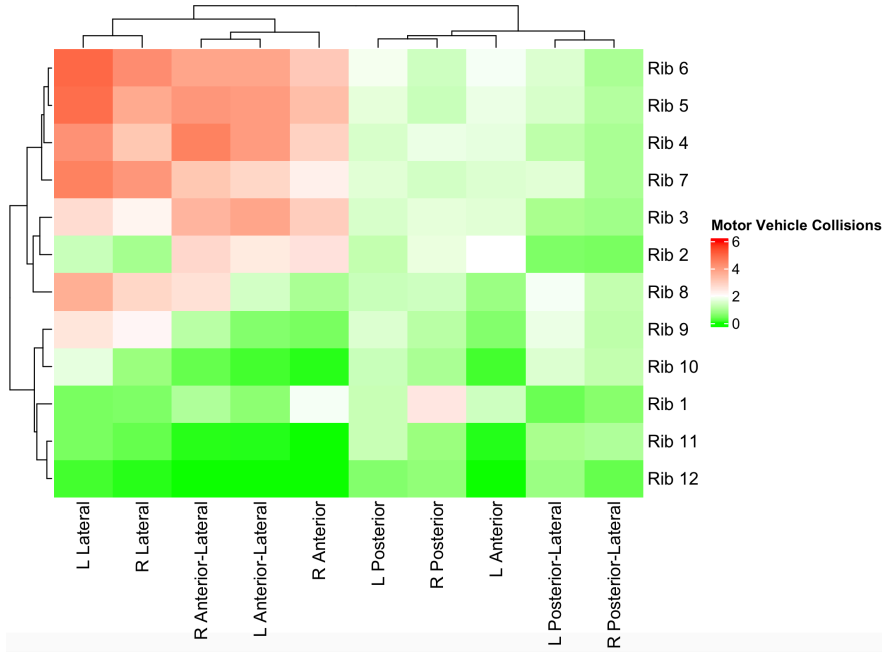
```
mvc_matrix_comb = cbind(mvc_Rmatrix, mvc_Lmatrix)
mvc_matrix_comb
```

#	R Posterior	R Posterior-Lateral	R Lateral	R Anterior-Lateral	R Anterior
## Rib 1	2.432	0.593	0.534	1.008	1.898
## Rib 2	1.720	0.474	0.890	2.788	2.550
## Rib 3	1.661	0.830	2.195	3.499	2.966
## Rib 4	1.779	0.949	3.144	4.508	2.906
## Rib 5	1.305	1.068	3.737	4.093	3.321
## Rib 6	1.364	0.949	4.270	3.796	3.084
## Rib 7	1.423	0.949	4.093	3.144	2.313
## Rib 8	1.364	1.246	2.847	2.610	0.949
## Rib 9	1.127	1.186	2.135	1.127	0.474
## Rib 10	0.949	1.246	0.771	0.356	0.059
## Rib 11	0.771	1.008	0.356	0.059	0.000
## Rib 12	0.712	0.356	0.059	0.000	0.000
##	L Anterior	L Anterior-Lateral	L Lateral	L Posterior-Lateral	L Posterior
## Rib 1	1.369	0.635	0.489	0.391	1.271
## Rib 2	2.004	2.395	1.320	0.538	1.222
## Rib 3	1.613	3.861	2.688	0.929	1.466
## Rib 4	1.711	4.008	4.203	1.173	1.466
## Rib 5	1.760	4.008	4.839	1.466	1.662
## Rib 6	1.906	3.861	4.985	1.515	1.857
## Rib 7	1.564	2.835	4.497	1.613	1.613
## Rib 8	0.782	1.417	3.666	1.906	1.320
## Rib 9	0.587	0.587	2.493	1.760	1.564
## Rib 10	0.147	0.147	1.711	1.564	1.320
## Rib 11	0.049	0.049	0.489	0.929	1.271
## Rib 12	0.000	0.000	0.147	0.782	0.587

Creating Heatmaps

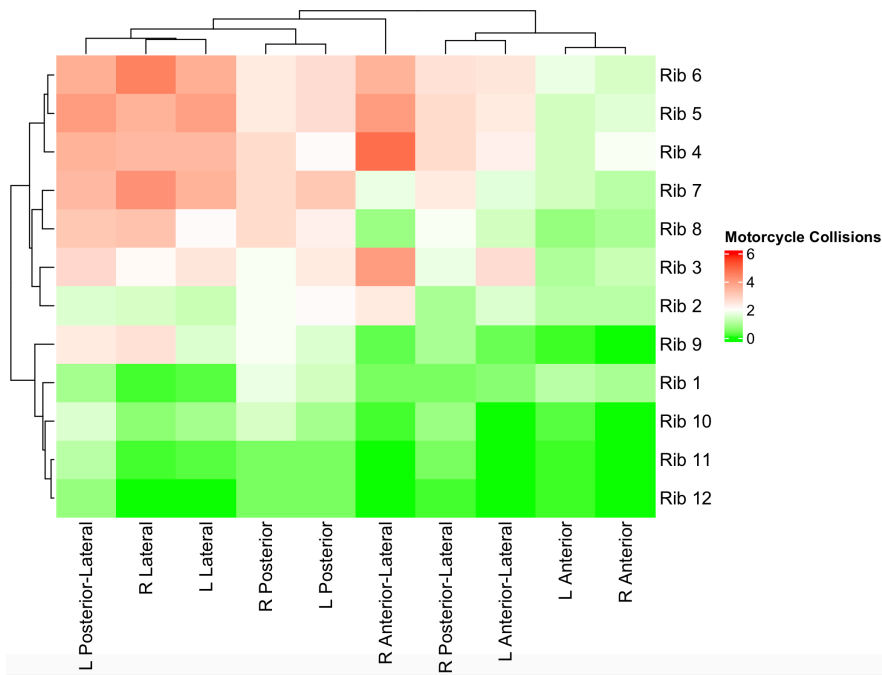
Heatmap for Motor Vehicle Collisions

```
col_fun = colorRamp2(c(0, 2, 6), c("green", "white", "red"))  
Heatmap(mvc_matrix_comb, name = "Motor Vehicle Collisions", col = col_fun)
```



Heatmap for Motorcycle Collisions

```
Heatmap(mcc_matrix_comb, name = "Motorcycle Collisions", col = col_fun)
```



Clustering, Dendrograms, & BIC

Motorcycle collisions

```
mcc_matrix_clust <- as.data.frame(t(scale(mcc_matrix_comb)))
mcc_clust_res = Mclust(mcc_matrix_clust)
mcc_clust_res$BIC

## Bayesian Information Criterion (BIC):
##      EII      VII      EEI      VEI      EVI      VVI
## 1 -138.19952 -138.1995 -135.51997 -135.5200 -135.5200 -135.520
## 2 -105.63634 -107.9386 -117.42669 -119.5318 -115.3381 -117.271
## 3  -97.06395      NA  -93.66867      NA      NA      NA
## 4 -104.56547      NA  -81.27941      NA      NA      NA
## 5  -80.91804      NA  -63.47364      NA      NA      NA
## 6  -78.18473      NA  -40.59207      NA      NA      NA
## 7  -50.44260      NA   64.86494      NA      NA      NA
## 8   33.96218      NA  140.68320      NA      NA      NA
## 9  121.42388      NA  254.25730      NA      NA      NA
##
## Top 3 models based on the BIC criterion:
##      EEI,9      EEI,8      EII,9
## 254.2573 140.6832 121.4239

mcc_dendro = as.dendrogram(hclust(dist(mcc_matrix_clust)))
```

The optimal number of clusters for the two-axis heatmap is 9, with the EEI clustering method.

Motor vehicle collisions

```
mvc_matrix_clust <- as.data.frame(t(scale(mvc_matrix_comb)))
mvc_clust_res = Mclust(mvc_matrix_clust)
mvc_clust_res$BIC

## Bayesian Information Criterion (BIC):
##      EII      VII      EEI      VEI      EVI      VVI
## 1 -262.42766 -262.4277 -262.168035 -262.1680 -262.1680 -262.1680
## 2 -220.69295 -221.3251 -221.204228 -207.8634 -216.9741 -196.4745
## 3 -199.05454 -195.1909 -171.924237 -175.6157 -161.4013 -161.1071
## 4 -149.32098      NA -139.467856      NA      NA      NA
## 5 -121.46812      NA -109.356630      NA      NA      NA
## 6  -96.64725      NA  -40.206551      NA      NA      NA
## 7   11.83499      NA   1.210773      NA      NA      NA
## 8  -20.44075      NA   55.340138      NA      NA      NA
## 9  139.12561      NA  207.778969      NA      NA      NA
##
## Top 3 models based on the BIC criterion:
##      EEI,9      EII,9      EEI,8
## 207.77897 139.12561 55.34014

mvc_dendro = as.dendrogram(hclust(dist(mvc_matrix_clust)))
```

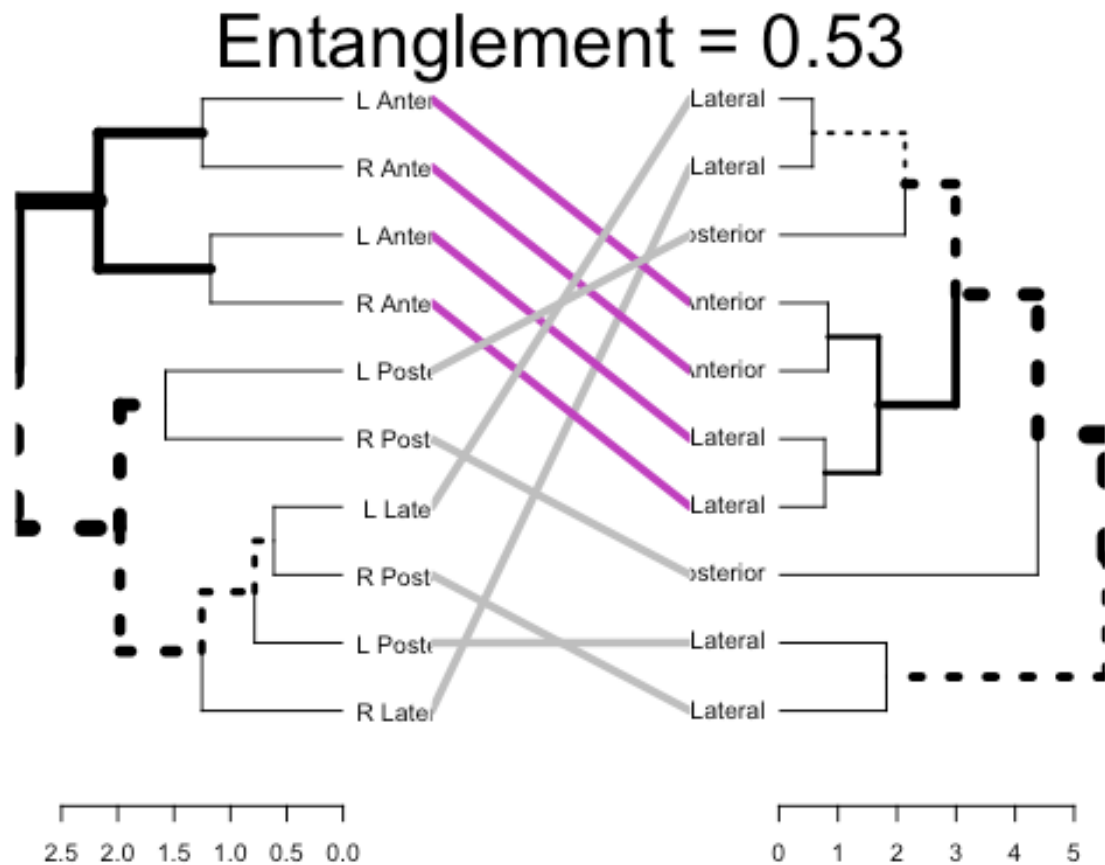
The optimal number of clusters for the two-axis heatmap is 9, with the EEI clustering method.

Comparing Two Dendrograms - 'Tangle-gram'

```
mcc_mvc_dendro = dendlist(mcc_dendro, mvc_dendro)  
tanglegram(mcc_mvc_dendro)
```

```
mcc_mvc_dendro %>% plot(main = paste("Entanglement =", round(entanglement(mcc_mvc_dendro), 2)))
```

MCC on LEFT, MVC on RIGHT



→ To cluster by the other axis, Rib-wise in this case, recreate the original matrix but only add formal labels to the 'rows' of Rib Number. Then create dendrograms using the 'as.dendrogram' function in dendextend

```
mcc_Lmatrix.1 = cbind(mcc_L_ant, mcc_L_antlat, mcc_L_lat, mcc_L_postlat, mcc_L_post)
rownames(mcc_Lmatrix.1) = c("Rib 1", "Rib 2", "Rib 3", "Rib 4", "Rib 5", "Rib 6", "Rib 7", "Rib 8", "Rib 9", "Rib 10", "Rib 11", "Rib 12")
mvc_Lmatrix.1 = cbind(mvc_L_ant, mvc_L_antlat, mvc_L_lat, mvc_L_postlat, mvc_L_post)
rownames(mvc_Lmatrix.1) = c("Rib 1", "Rib 2", "Rib 3", "Rib 4", "Rib 5", "Rib 6", "Rib 7", "Rib 8", "Rib 9", "Rib 10", "Rib 11", "Rib 12")

mccL_1.dendro = as.dendrogram(hclust(dist(mcc_Lmatrix.1)))
mvcL_1.dendro = as.dendrogram(hclust(dist(mvc_Lmatrix.1)))
```

Comparing Two Dendrograms - Baker's Gamma Index

H0 distribution under Baker's Gamma Curve

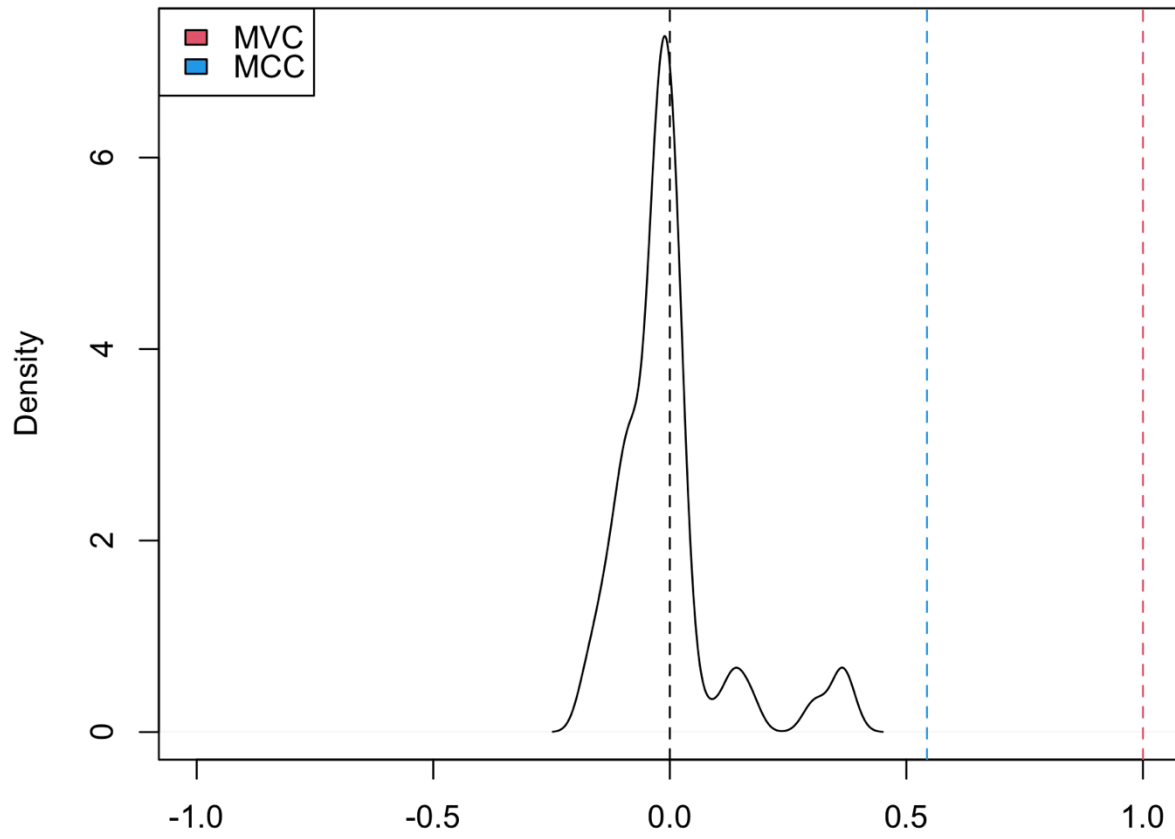
```
mvc_bgi.L = cor_bakers_gamma(mvcL_1.dendro, mvcL_1.dendro)
mcc_bgi.L = cor_bakers_gamma(mvcL_1.dendro, mccL_1.dendro)

R <- 100
L_bgi_coll_results = numeric(R)
L.coll.dendro_bgi = mvcL_1.dendro
for(i in 1:R) {
  L.coll.dendro_bgi <- sample.dendrogram(L.coll.dendro_bgi, replace = FALSE)
  L_bgi_coll_results[i] <- cor_bakers_gamma(mvcL_1.dendro, L.coll.dendro_bgi)
}
plot(density(L_bgi_coll_results),
     main = "Baker's gamma distribution under H0: L-side MVC vs. MCC",
     xlim = c(-1,1))
abline(v = 0, lty = 2)
abline(v = mvc_bgi.L, lty = 2, col = 2)
abline(v = mcc_bgi.L, lty = 2, col = 4)
legend("topleft", legend = c("MVC", "MCC"), fill = c(2,4))
round(sum(mcc_bgi.L < L_bgi_coll_results)/ R, 4)

## [1] 0

title(sub = paste("One sided p-value:",
                 "MVC =", round(sum(mvc_bgi.L < L_bgi_coll_results)/ R, 4),
                 " ; MCC =", round(sum(mcc_bgi.L < L_bgi_coll_results)/ R, 4)
))
```

Baker's gamma distribution under H0: L-side MVC vs. MCC



N = 100 Bandwidth = 0.02219
One sided p-value: MVC = 0 ; MCC = 0