Distinct patterns for zeroes in Euler diagrams on three sets

Jonathan Swinton

1st August, 2009

This vignette does not need to be read to understand the Vennerable package.

The Euler diagram on three sets has seven regions which are not dark matter. If the weight associated to the region is zero, we do not want to display the region. The number of possible patterns of zeros is $2^7 = 128$ but many of these patterns are symmetric under a relabelling of the original sets.

How many distinct zero-patterns are there, allowing set relabelling?

```
> library(xtable)
> vs <- expand.grid(A = c(0, 1), B = c(0, 1), C = c(0, 1))
> vs$VS <- apply(data.matrix(vs[, 1:3]), 1, paste, collapse = "")
> vs <- subset(vs, VS != "000")
> Euler.from.vs <- function(vs) {
+     w <- lapply(vs$VS, function(x) {
+         c(0, 1)
+     })
+     names(w) <- vs$VS
+     Eulers <- do.call(expand.grid, w)
+     Eulers$VS <- apply(data.matrix(Eulers), 1, paste, collapse = "")
+     Eulers
+ }
> library(gtools)
> worder <- permutations(3, 3)
> worder <- lapply(1:nrow(worder), function(x) {
+     worder[x, ]
+ })
> P2 <- lapply(worder, function(x) {
+     wname <- paste("Order", paste(x, collapse = ""))
+     vs.order <- vs[, x]
+     E2 <- Euler.from.vs(vs)
+     E2 <- E2[do.call(order, E2[, 1:7]), ]
+     vs.order$VS <- apply(data.matrix(vs.order), 1, paste, collapse = "")
+     vs.perm <- match(vs.order$VS, vs$VS)
+     E2.perm <- E2[, vs.perm]
+     E2.perm$VS <- apply(data.matrix(E2.perm), 1, paste, collapse = "")
+     E2.perm$VS
+ })
> E3 <- do.call(rbind, P2)
```

```
> F3 <- unique(apply(E3, 2, function(x) (unique(sort(x)))))
> iclasses <- (sapply(F3, paste, collapse = ";"))
> rclasses <- sapply(F3, function(x) x[1])
> irclasses <- data.frame(VS = rclasses, iclasses = iclasses, stringsAsFactors = FALSE)
> E1 <- Euler.from.vs(vs)
> Eclass <- merge(E1, irclasses)
> rownames(Eclass) <- 1:nrow(Eclass)
> Eclass <- Eclass[order(Eclass$VS), ]
```

However some of these (eg 0000010) correspond to patterns in which every region at least one set is empty.

```
> vsnames <- names(E1)[1:7]
> vsmat <- do.call(rbind, strsplit(vsnames, split = ""))
> isa <- vsnames[vsmat[, 1] == "1"]
> isb <- vsnames[vsmat[, 2] == "1"]
> isc <- vsnames[vsmat[, 3] == "1"]
> havea <- apply(Eclass[, isa], 1, sum) > 0
> haveb <- apply(Eclass[, isb], 1, sum) > 0
> havec <- apply(Eclass[, isc], 1, sum) > 0
> Ehave <- Eclass[havea & haveb & havec, ]
> rownames(Ehave) <- 1:nrow(Ehave)
```

There are 34 patterns with all sets represented

```
> print(xtable(Ehave, digits = 0), size = "small")
```

[1]

References

[1] A. W. F. Edwards. *Cogwheels of the Mind: The Story of Venn Diagrams*. The John Hopkins University Press, Baltimore, Maryland, 2004.

	VS	100	010	110	001	101	011	111	iclasses
1	0000001	0	0	0	0	0	0	1	0000001
2	0000011	0	0	0	0	0	1	1	0000011;0000101;0010001
3	0000110	0	0	0	0	1	1	0	0000110;0010010;0010100
4	0000111	0	0	0	0	1	1	1	0000111;0010011;0010101
5	0001001	0	0	0	1	0	0	1	0001001;0100001;1000001
6	0001011	0	0	0	1	0	1	1	0001011;0001101;0100011;0110001;1000101;1010001
7	0001110	0	0	0	1	1	1	0	0001110;0110010;1010100
8	0001111	0	0	0	1	1	1	1	0001111;0110011;1010101
9	0010110	0	0	1	0	1	1	0	0010110
10	0010111	0	0	1	0	1	1	1	0010111
11	0011000	0	0	1	1	0	0	0	0011000;0100100;1000010
12	0011001	0	0	1	1	0	0	1	0011001;0100101;1000011
13	0011010	0	0	1	1	0	1	0	0011010;0011100;0100110;0110100;1000110;1010010
14	0011011	0	0	1	1	0	1	1	0011011;0011101;0100111;0110101;1000111;1010011
15	0011110	0	0	1	1	1	1	0	0011110;0110110;1010110
16	0011111	0	0	1	1	1	1	1	0011111;0110111;1010111
17	0101001	0	1	0	1	0	0	1	0101001;1001001;1100001
18	0101011	0	1	0	1	0	1	1	0101011;1001101;1110001
19	0101100	0	1	0	1	1	0	0	0101100;0111000;1001010;1011000;1100010;1100100
20	0101101	0	1	0	1	1	0	1	0101101;0111001;1001011;1011001;1100011;1100101
21	0101110	0	1	0	1	1	1	0	0101110;0111010;1001110;1011100;1110010;1110100
22	0101111	0	1	0	1	1	1	1	0101111;0111011;1001111;1011101;1110011;1110101
23	0111100	0	1	1	1	1	0	0	0111100;1011010;1100110
24	0111101	0	1	1	1	1	0	1	0111101;1011011;1100111
25	0111110	0	1	1	1	1	1	0	0111110;1011110;1110110
26	0111111	0	1	1	1	1	1	1	0111111;1011111;1110111
27	1101000	1	1	0	1	0	0	0	1101000
28	1101001	1	1	0	1	0	0	1	1101001
29	1101010	1	1	0	1	0	1	0	1101010;1101100;1111000
30	1101011	1	1	0	1	0	1	1	1101011;1101101;1111001
31	1101110	1	1	0	1	1	1	0	1101110;1111010;1111100
32	1101111	1	1	0	1	1	1	1	1101111;1111011;1111101
33	1111110	1	1	1	1	1	1	0	1111110
34	1111111	1	1	1	1	1	1	1	1111111