A Comparison of R and SPRINT hamming.distance functions

# Summary

Hamming and phamming do different things at the moment. Hamming takes in a binary matrix and returns a matrix of the hamming distance between each of the rows in the matrix. Phamming takes in ShortReadQ or DNAStringSet objects and writes the hamming distance between each of the DNA strings to a binary file. Phamming compares character strings instead of 1s and 0s.

Biological data would have to be pre and post- processed to use the R version of hamming. I could do this, and have a way to map DNA to binary, but maybe it’s worth checking how slow hamming actually is first.

# R library e1071’s hamming.distance function

This takes in 2 vectors containing logical values (true/false) and compares them, outputting the number of positions that differ between them. Alternatively, it takes in a matrix and compares the rows, all against all, as above.

From the manual:

Description

If both x and y are vectors, hamming.distance returns the Hamming distance (number of diﬀerent bytes) between this two vectors. If x is a matrix, the Hamming distances between the rows of x are computed and y is ignored.

Usage

hamming.distance(x, y)

Arguments

x a vector or matrix.

y an optional vector.

Examples

> x <- c(1, 0, 0)

> y <- c(1, 0, 1)

> hamming.distance(x, y)

[1] 1

> z <- rbind(x,y)

> rownames(z) <- c("Fred", "Tom")

> hamming.distance(z)

Fred Tom

Fred 0 1

Tom 1 0

Code

The code’s short, so I’ll reproduce it here:

> hamming.distance

function (x, y)

{

z <- NULL

if (is.vector(x) && is.vector(y)) {

z <- sum(as.logical(x) != as.logical(y))

}

else {

z <- matrix(0, nrow = nrow(x), ncol = nrow(x))

for (k in 1:(nrow(x) - 1)) {

for (l in (k + 1):nrow(x)) {

z[k, l] <- hamming.distance(x[k, ], x[l, ])

z[l, k] <- z[k, l]

}

}

dimnames(z) <- list(dimnames(x)[[1]], dimnames(x)[[1]])

}

z

}

# SPRINT phamming.distance

Description

Calculates the hamming distance matrix between a set of DNA

sequences provided as a ShortReadQ or DNAStringSet objects.

Usage

phamming.distance(data, output\_filename)

Arguments

data: ShortReadQ or DNAStringSet objects

output\_filename: results will be stored here as binary data

Example

> a <- readFastq("data.fastq")

> phamming.distance(a, "phamming\_result.out")

chunk\_size: 10

[[1]]

[1] "AACAGTGTCATTTGATGACCCTCCTTCAGCTAAATGCTCTTCCCACCTTCAACACTAACCAAAAATTCAATGTGTTTATAAGCCAAACAGTCACTTAAAAAAATGCTGGGCGCCGTGGCTCACGCCTGAAAGAAGACCTTACCTCCACTAAATGGAGGACACCCACTCCACCTTACTACCAGACAACCTTAGATCAAATTAAAATTGAACAGATTAAATTTGATAGAGAACATACATACTCTTAATAACTTCAACCATGAGTTTTGTGCTTCCTGTCCAACTTTTCTTAAGGAGGTTCCTTCTGTGCCAGAAACCCTTAA"

[[2]]

[1] "phamming\_result.out"

[[3]]

[1] 32

[[4]]

[1] 10

Outputs: the input data, the output filename, the sample width and the number of samples. The hamming distances for each row vs row comparison are output to a binary file.