

# Halton Iterative Partitioning (HIP) Sampling for Points

*Aidan McDonald*

*October 29, 2018*

## Introduction

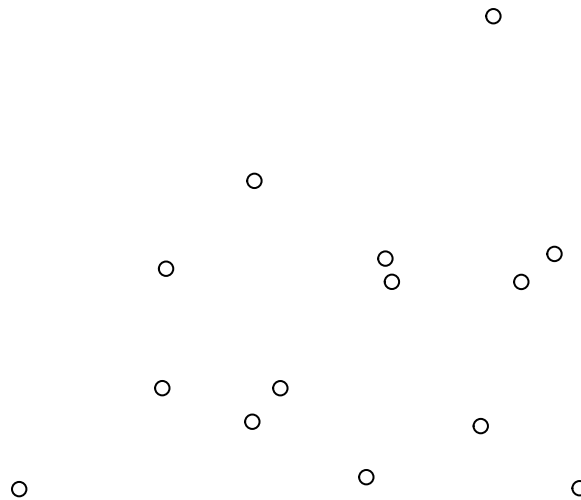
This vignette demonstrates basic Halton Iterative Partitioning (HIP) sampling from a point resource.

```
library(SDraw)
```

## Small Sample

First, we draw a HIP sample of 5 points from a resource of 14 points. This example demonstrates how HIP sampling from point resources works. The next plot is 14 points drawn from a standard bivariate normal distribution.

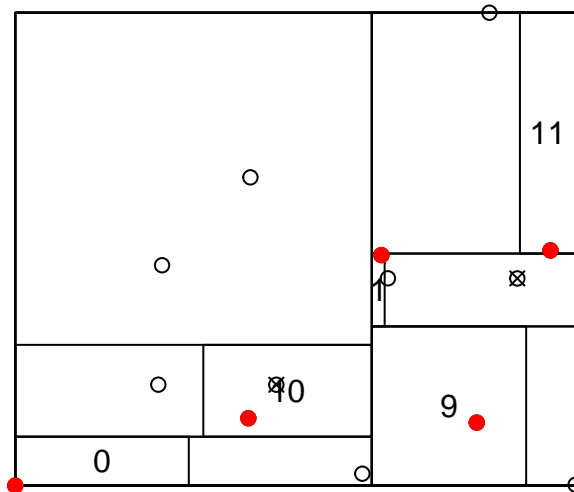
```
n <- 5  
N <- 14  
x <- rnorm(N,0,1)  
y <- rnorm(N,0,1)  
sp <- SpatialPoints(cbind(x,y))  
plot(sp, pch=1)
```



HIP sampling first constructs a Halton lattice around the points such that the inclusion probability inside all boxes is the same. In this case, all points are equiprobable and each box in the lattice will contain the same number of points. To do this, some points may be randomly discarded from the potential sample if the total is not divisible by 2 and 3. Then, a sequence of  $n$  consecutive Halton indices is selected and a point from each of the corresponding Halton boxes is sampled.

Note that in SDraw's built-in "plot.lattice" option for HIP sampling, sampled points are shown in red, the indices of the selected Halton boxes are displayed in the boxes, and discarded points are marked with an X. Also, it is possible that not all halton boxes will be drawn because it is often not necessary to delineate every box in the lattice.

```
s <- sdraw(sp, n, "HIP", plot.lattice=TRUE)
```



## Larger Sample

Now, we implement HIP sampling on a larger point resource, the cities in the state of Washington.

```
data(WA.cities)
n <- 150
s <- sdraw(WA.cities, n, "HIP")
plot(WA)
points(WA.cities)
points(s, col="red", pch=19)
```

