

```

1  ### generate.bootstrap.p.value.R #####
2  # estimate probability of overlap of three datasets vs. expected by chance alone
3
4  ### FUNCTIONS #####
5  # function that carries out bootstrap p-value generation for 1 permutation
6  hypergeometric.data.generate <- function(list.of.sample.lengths, total.length=10445) {
7      true.positions <- lapply(
8          list.of.sample.lengths,
9          function(sample.length) {
10              return(
11                  sample(
12                      x = total.length,
13                      size = sample.length
14                  )
15              );
16          }
17      );
18      table.positions <- table(unlist(true.positions));
19      common.positions <- table.positions[table.positions==3];
20      return(length(common.positions));
21  }
22
23  ### MAIN #####
24  # read in main results dataframe
25  results.dataframe <- read.table(
26      file = '2014-08-07_Constitutive.AHR_common.annotated.fit.tissues.txt',
27      header = TRUE,
28      sep = "\t",
29      quote = "",
30      comment.char = "",
31      fill = TRUE
32  );
33
34  # clean up dataframe to only keep the columns needed for this analysis
35  subset.results <- results.dataframe[, grep('Q.|HID', names(results.dataframe))];
36
37  ### SET LENGTHS #####
38  # establish lists of significant genes
39  rat.liver.sig.genes <- subset.results$HID[
40      subset.results[greps('Rat.Liver', names(subset.results))] < 0.05
41  ];
42  rat.adipose.sig.genes <- subset.results$HID[
43      subset.results[greps('Rat.Adipose', names(subset.results))] < 0.05
44  ];
45  rat.hypothalamus.sig.genes <- subset.results$HID[
46      subset.results[greps('Rat.Hypothalamus', names(subset.results))] < 0.05
47  ];
48
49  # count how many overlap in all three datasets
50  observed.overlap <- length(
51      intersect(
52          intersect(
53              rat.liver.sig.genes,
54              rat.adipose.sig.genes
55          ),
56          rat.hypothalamus.sig.genes
57      )
58  );
59
60  # initiate list of genes
61  gene.lengths <- list(
62      length(rat.liver.sig.genes),
63      length(rat.adipose.sig.genes),
64      length(rat.hypothalamus.sig.genes)
65  );
66
67  # set number of permutations
68  permutations <- 1000000;
69
70  # apply bootstrap, hypergeometric mimic:

```

```
71 #      find occurrences of overlap counts >= observed count
72 counts.of.overlaps <- replicate(
73     n = permutations,
74     expr = hypergeometric.data.generate(gene.lengths)
75 );
76 frequency.greater <- length(
77     which(counts.of.overlaps >= observed.overlap)
78 ); # 0 occurrences of overlap > 28, highest is 9
79
80 # estimated p-value from permutations
81 p.value.estimate <- frequency.greater/ permutations;
82 print(p.value.estimate);
83
84 # Result after running code:
85 #      p.value.estimate = 0
86 #      therefore, p < 1/1,000,000
```