IS 622 Homework Week 7

Ben Arancibia

October 10, 2015

6.1.1

Suppose there are 100 items, numbered 1 to 100, and also 100 baskets, also numbered 1 to 100. Item i is in basket b if and only if i divides b with no remainder. Thus, item 1 is in all the baskets, item 2 is in all fifty of the even-numbered baskets, and so on. Basket 12 consists of items {1, 2, 3, 4, 6, 12}, since these are all the integers that divide 12.

Answer the following questions:

(a) If the support threshold is 5, which items are frequent?

```
factors <- function(n)
{
   if(length(n) > 1)
   {
      lapply(as.list(n), factors)
   } else
   {
      one.to.n <- seq_len(n)
      one.to.n[(n %% one.to.n) == 0]
   }
}
x <- factors(c(1:100))
list <- x[lapply(x, length) > 5]
```

36 Frequent items are: 12, 18, 30, 24, 28, 30, 32, 36, 40, 42, 44, 45, 48, 50, 52, 54, 56, 60, 63, 64, 66, 68, 70, 72, 75, 76, 78, 80, 84, 88, 90, 92, 96, 98, 99, 100

(b) If the support threshold is 5, which pairs of items are frequent?

```
library(plyr)
```

```
## Warning: package 'plyr' was built under R version 3.1.3
```

```
df <- ldply(list, data.frame)
```

(1,2) (1,3) (2,3) (1,6)

(c) What is the sum of the sizes of all the baskets?

```
unlist(lapply(list, function(x) sum(x)))
```

```
##  [1]  28  39  42  60  56  72  63  91  90  96  84  78 124  93  98 120 120
## [18] 168 104 127 144 126 144 195 124 140 168 186 224 180 234 168 252 171
## [35] 156 217
```

6.1.5

For the data of Exercise 6.1.1, what is the confidence of the following association rules?

(a) {5,7} -> 2.

{5,7} appears in 2 basket. 2 is in the same basket so it would be 1/2

(b) {2,3,4} -> 5.

{2,3,4} appeasr 7 times and 5 appears in 1 of those baskets. The confidence is 1/8

6.3.1

Here is a collection of twelve baskets. Each contains three of the six items 1 through 6.

{1,2,3} {2,3,4} {3,4,5} {4,5,6} {1,3,5} {2,4,6} {1,3,4} {2,4,5} {3,5,6} {1,2,4} {2,3,5} {3,4,6}

Suppose the support threshold is 4. On the first pass of the PCY Algorithm we use a hash table with 11 buckets, and the set {i, j} is hashed to bucket i × j mod 11.

(a) By any method, compute the support for each item and each pair of items.

Take the list of baskets and apply the same algorithm as in 6.1.1.

for b in baskets if (i%in% b)

Baskets are : {1,2,3} {2,3,4} {3,4,5} {4,5,6} {1,3,5} {2,4,6} {1,3,4} {2,4,5} {3,5,6} {1,2,4} {2,3,5} {3,4,6}

```
for (i in 1:6) {
  basketcount <- 0
  for (b in baskets) {
    if (i %in% b) {
      basketcount <- basketcount + 1}
}
print(paste(i, count))
}
```

Baskets	Count
1	4
2	6
3	8
4	8
5	6
6	4

Count which baskets support each bucket

Pairs	Support
(1,2)	2

Pairs	Support
(1,3)	3
(1,4)	2
(1,5)	1
(1,6)	0
(2,3)	3
(2,4)	4
(2,5)	2
(2,6)	1
(3,4)	4
(3,5)	4
(3,6)	2
(4,5)	3
(4,6)	3
(5,6)	2

(b) Which pairs hash to which buckets?

FOR (each basket) { FOR (each item) add 1 to item's count; FOR (each pair of items) { hash the pair to a bucket (1); add 1 to the count for that bucket } }

```
pairs <- matrix(c(1,1,2,1,2,3,1,2,3,4,1,2,3,4,5, 2,3,3,4,4,4,5,5,5,5,6,6,6,6,6), nrow=15,ncol=2)

for (i in 1:nrow(pairs)) {
  hash <- (pairs[i,1] * pairs[i,2]) %% 11
  numbers <- paste("(",pairs[i,1], ",", pairs[i,2], ")", sep="")
  print(paste(numbers,hash))}
```

```
## [1] "(1,2) 2"
## [1] "(1,3) 3"
## [1] "(2,3) 6"
## [1] "(1,4) 4"
## [1] "(2,4) 8"
## [1] "(3,4) 1"
## [1] "(1,5) 5"
## [1] "(2,5) 10"
## [1] "(3,5) 4"
## [1] "(4,5) 9"
## [1] "(1,6) 6"
## [1] "(2,6) 1"
## [1] "(3,6) 7"
## [1] "(4,6) 2"
## [1] "(5,6) 8"
```

(c) Which buckets are frequent?

For the buckets calculated above, 1,2,4,6, and 8 are frequent with 2 pairs associated with each bucket.

(d) Which pairs are counted on the second pass of the PCY Algorithm?

For the pairs to be counted in the second pass of the PCY Algorithm, the support has to be 4 or greater (our initial threshold is 4). Pair(3,4), Pair(3,5), and Pair(2,4) have support that are equal to four. This can be seen above in part (a) when calculating pairs and support.

6.3.2

Suppose we run the Multistage Algorithm on the data of Exercise 6.3.1, with the same support threshold of 4. The first pass is the same as in that exercise, and for the second pass, we hash pairs to nine buckets, using the hash function that hashes {i, j} to bucket i + j mod 9. Determine the counts of the buckets on the second pass. Does the second pass reduce the set of candidate pairs? Note that all items are frequent, so the only reason a pair would not be hashed on the second pass is if it hashed to an infrequent bucket on the first pass.

The first step is take all the pairs and then pass through them. For the second pass hash pairs to nine buckets instead of 11.

Start with the Buckets in (b) in 6.3.1.

(2,6), (3,4), (1,2), (4,6), (1,4), (3,5), (1,6), (2,3), (2,4), (5,6)

then apply same logic but into 9 buckets

```
FOR (each basket) {
  FOR (each item)
    add 1 to item's count;
  FOR (each pair of items) {
    hash the pair to a bucket (9);
    add 1 to the count for that
    bucket
  }
}
```

```
pairs <- matrix(c(1,1,2,1,2,3,1,2,3,4,1,2,3,4,5, 2,3,3,4,4,4,5,5,5,5,6,6,6,6,6), nrow=15,ncol=2)

for (i in 1:nrow(pairs)) {
  hash <- (pairs[i,1] * pairs[i,2]) %% 9
  numbers <- paste("(",pairs[i,1], ",", pairs[i,2], ")", sep="")
  print(paste(numbers,hash))}
```

```
## [1] "(1,2) 2"
## [1] "(1,3) 3"
## [1] "(2,3) 6"
## [1] "(1,4) 4"
## [1] "(2,4) 8"
## [1] "(3,4) 3"
## [1] "(1,5) 5"
## [1] "(2,5) 1"
## [1] "(3,5) 6"
## [1] "(4,5) 2"
## [1] "(1,6) 6"
## [1] "(2,6) 3"
## [1] "(3,6) 0"
## [1] "(4,6) 6"
## [1] "(5,6) 3"
```

“‘

Yes the second pass reduces the count of buckets because only have nine buckets instead of eleven