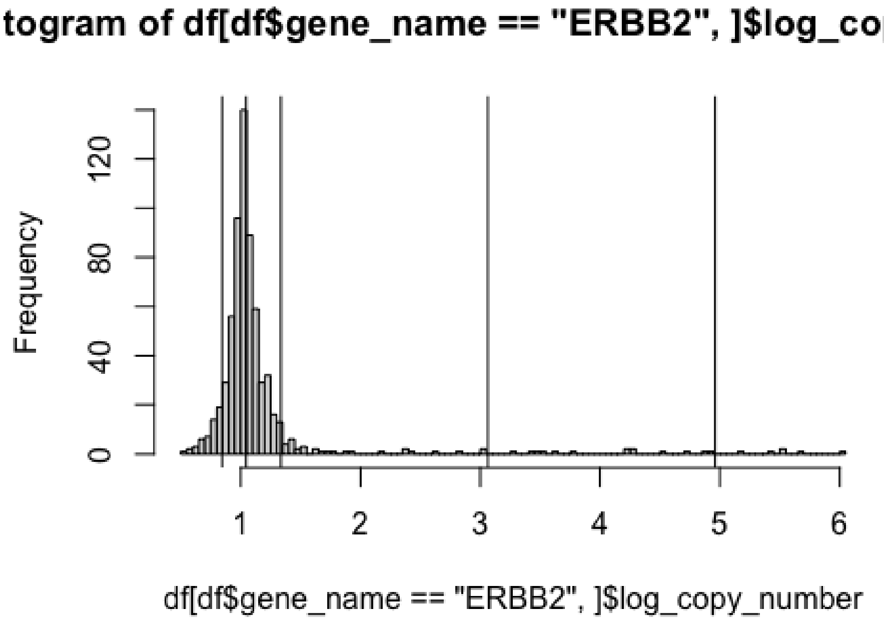
I think I've come up with a method to detect copy number distributions, but this needs some input to tweak. for example, gene ERBB2 can be seen as 2 peaks as when I plot it in histogram I saw the below (Ignore the vertical lines):



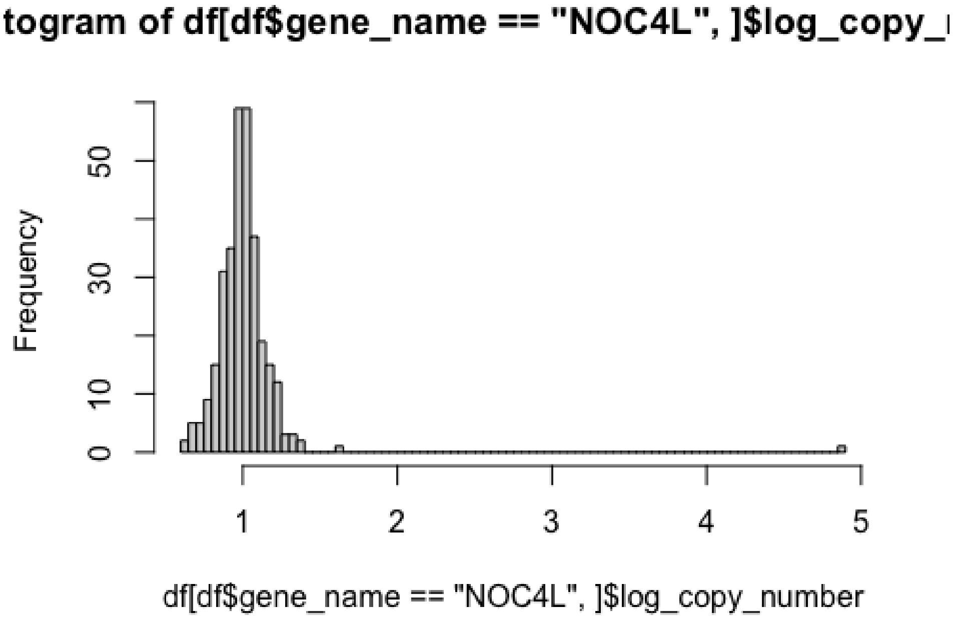
this is achieved by looking at an increased number in a bin, also, to eliminate noise, I used a cutoff of 1%, i.e. the bin needs to have at least 1% of observations to be able to have the mass to be called a peak. The first peak is obvious and the second peak is the small shoulder immediately right to the first peak (the bar at ~1.2). The actual numbers around each peaks are actually listed below, you can see first peak is at 140, second peak is going from 29 to 32 then drop to 16:

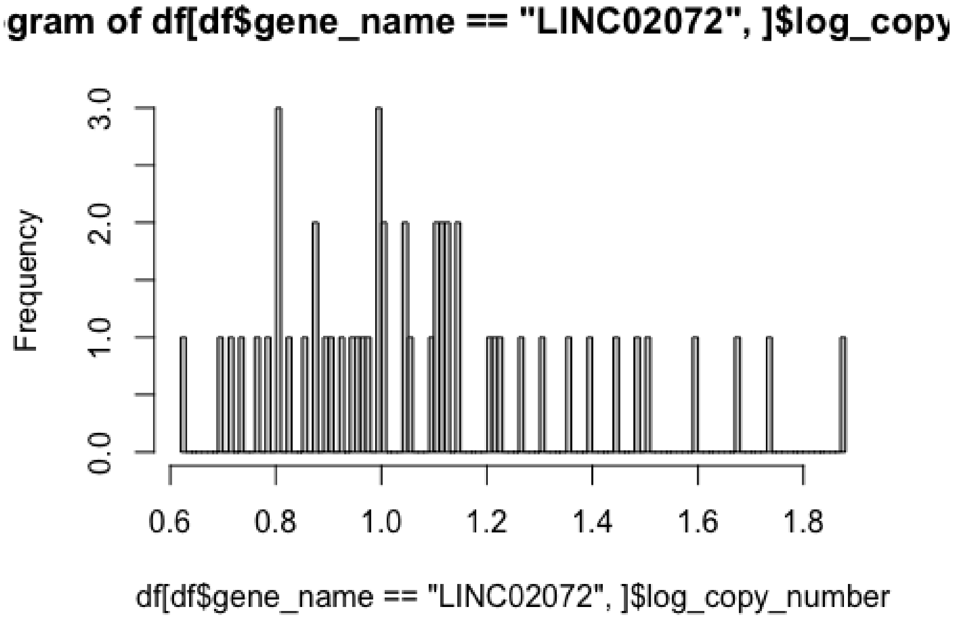
```
$counts
  [1]   1   2   3   6   7  14  19  29  56  96 140  89  59  29  32  16  13   4   6   2   3   0   2   1
 [25]   1   1   0   1   1   0   0   0   0   1   0   0   0   2   1   0   0   0   1   0   0   0   1   0
 [49]   0   0   2   0   0   0   0   1   0   0   1   1   1   0   1   0   0   1   0   0   0   0   0   0
 [73]   0   0   2   2   0   0   0   0   1   0   0   0   1   0   0   1   1   0   0   0   0   1   0   0
 [97]   0   0   1   0   2   0   0   1   0   0   0   0   0   0   1
```

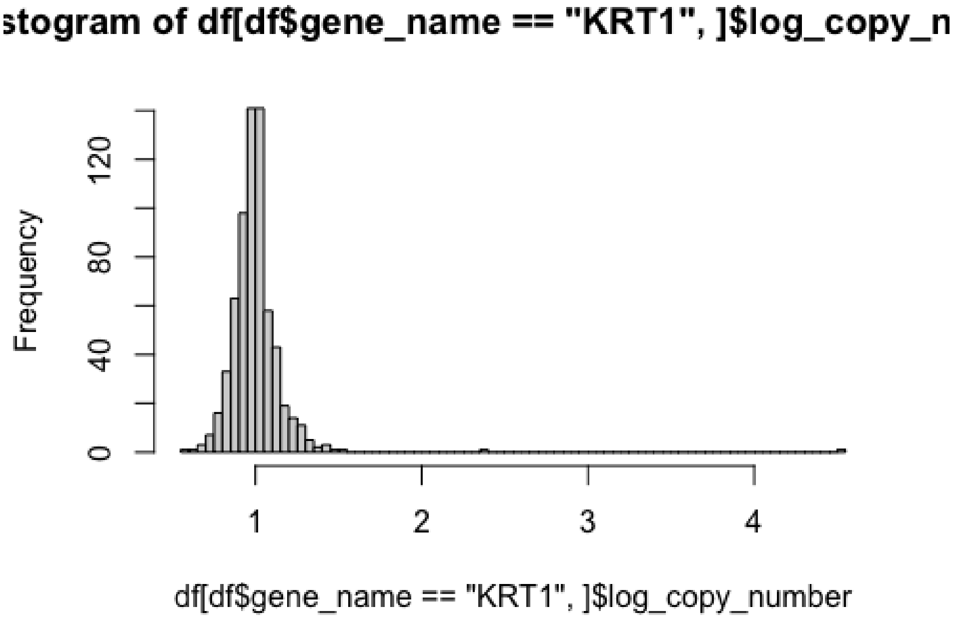
Using this method, I looked into the distribution of ~17k genes, simply ask how many peaks they have, and here is the distribution:

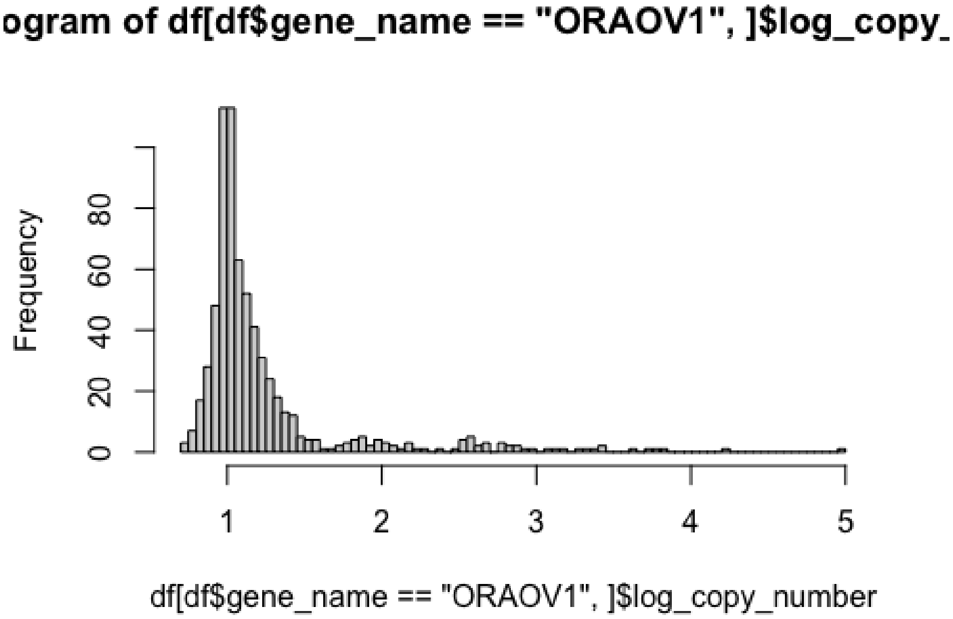
```
   0    1    2    3    4    5    6    7    8    9   10   11   12   13   14   15 
   4  272  356  490  991 1995 2800 2937 2615 1950 1264  674  256   75   11    2 
```

you can see that there are many many more peaks, but it does detect peaks mostly, only 4 doesn't have any peak, and they are shown below as you can see it is safe to say they have 1 peak:

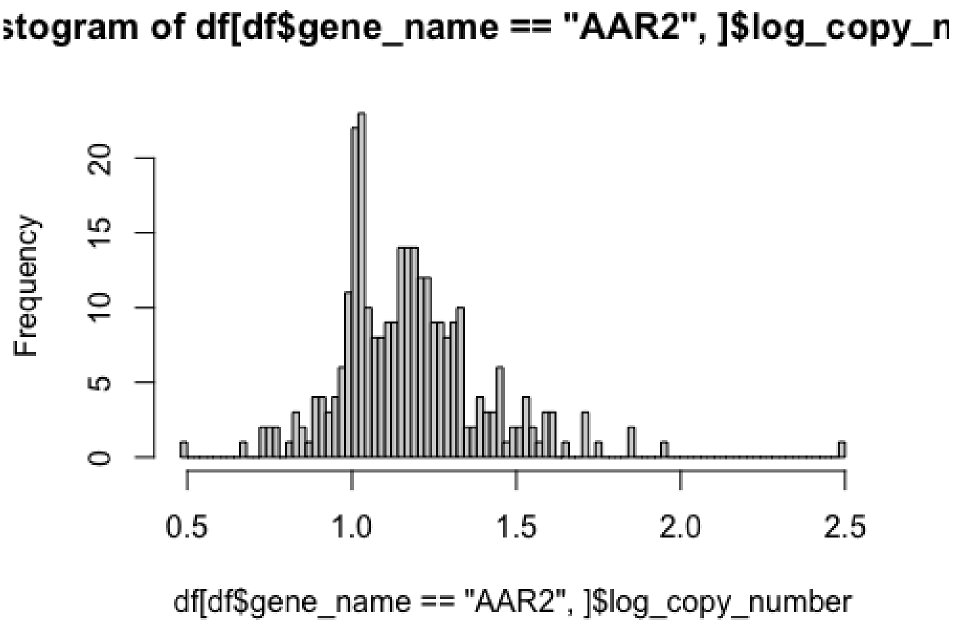

togram of df[df$gene_name == "NOC4L",]$log_copy_
Frequency
0
10
30
50
1
2
3
4
5
df[df$gene_name == "NOC4L",]$log_copy_number


gram of df[df$gene_name == "LINC02072",]$log_copy
Frequency
0.0
1.0
2.0
3.0
0.6
0.8
1.0
1.2
1.4
1.6
1.8
df[df$gene_name == "LINC02072",]$log_copy_number

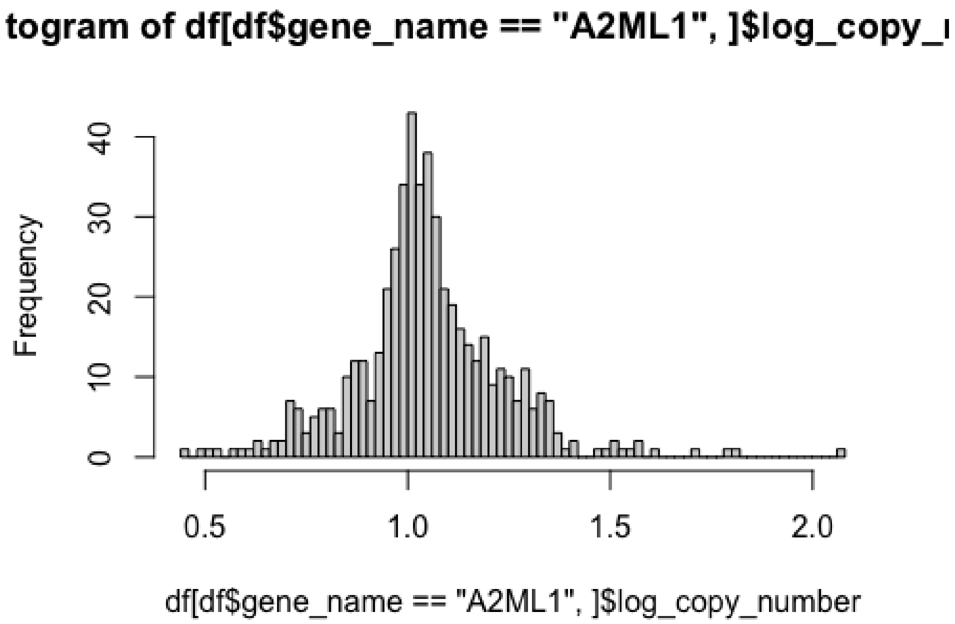




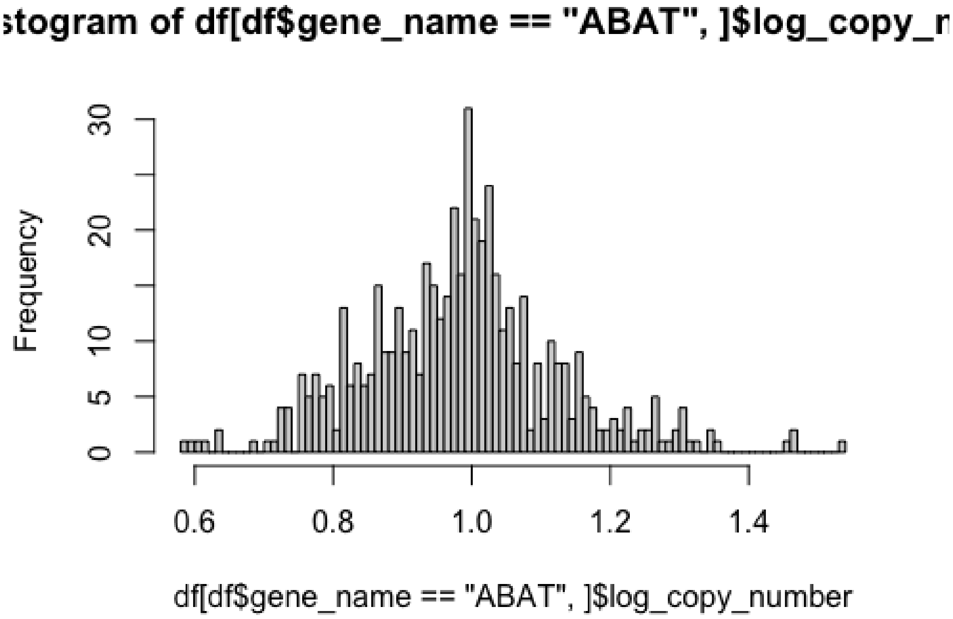
3 peaks genes looks like this:



6 peak genes like this:



15 peaks looks like this:



This makes me think I should increase 1% to 10% and maybe changed the breaks of the histogram to less granular.

The idea is then translate this peaking profile into features and feed them into the kmeans cluster (i.e. unsupervised machine learning) algorithm.