Chapter 3 Class Notes – Word Distributions and Occurrences

3.1. The Biological Problem: "restriction endonucleases provide[s] the means for precisely and reproducibly cutting the DNA into fragments of manageable size" and a "restriction map is a display of positions on a DNA molecule of locations of cleavage by one or more restriction endonucleases" – serves as a "fingerprint". Recall that "cloning puts DNA of manageable size into vectors that allow the inserted DNA to be amplified, and the reason for doing this is that large molecules cannot be readily manipulated without breakage." Our focus:

- **If we are able to digest (cut) the DNA with a restriction endonuclease such as *Eco*RI, approximately how many fragments would be obtained, and what would be their size distribution?**
- **Suppose that we observe 761 occurrences of the sequence 5'-GCTGGTGG-3' in a genome that is 50% G+C and 4.6 Mb in size. How does this number compare with the expected number, and expected according to what model?**

3.2 Modeling the Number of Restriction Sites in DNA: we know the (%G+C), and we'll assume (for now) *iid* letters (and that is a very big assumption!). Regarding the number of restriction sites, "our model is going to assume that cleavage can occur between any two successive positions on the DNA." Even though the X_i are not in fact independent, the "binomial distribution often works well nonetheless." To test the *iid* model, we compare actual with expected: lambda DNA sequence is 48,502 bp long,

and we use $p_A = p_T = 0.2507, p_C = p_G = 0.2493$:

Table 3.1. Comparison of observed and expected numbers of restriction enzyme cleavage sites for bacteriophage lambda DNA. (The data were taken from GenBank nucleotide sequence file NC_001416 and from the New England Biolabs online catalog (`http://www.neb.com`–Restriction Maps/Frequencies of Restriction Sites).

Enzyme	Recognition sequence	p	$\mathbb{E}N$	Var$N$	Observed number
*Alu*I	AGCT	0.00391	190	189	143
*Bfa*I	CTAG	0.00391	190	189	13*
*Bst*UI	CGCG	0.00386	187	186	157
*Hae*III	GGCC	0.00386	187	186	149
*Hpa*II	CCGG	0.00386	187	186	328*
*Mbo*I	GATC	0.00391	190	189	116*
*Mse*I	TTAA	0.00395	192	191	195
*Nla*III	CATG	0.00391	190	189	181
*Rsa*I	GTAC	0.00391	190	189	113*
*Taq*I	TCGA	0.00391	190	189	121*

In most cases, this model fits well: "suggesting that the *iid* model adequately describes the number of restriction sites for lambda DNA. There are five enzymes whose site numbers in lambda are well over three standard deviations away from the predicted value. If this were consistently observed for other bacteriophages of *E. coli* and for the *E. coli* chromosome, then we might hypothesize that the deficiency of these recognition sequences may reflect some biochemical feature of the organism (e.g., peculiarities of the DNA repair system)."

3.2.4. Poisson Approximation to the Binomial Distribution: hello

Suppose that N has a Binomial distribution with parameters n and p. Then, since the mean is np and the variance is $np(1-p)$,

whenever p is very small, the two are nearly the same. It's not hard to show that for large n and small p (math language: as $n \to \infty$ and $p \to 0$ so that $\lambda = np$ stays constant), then the distribution for N is approximately that of a Poisson RV:

$$P(N = j) = \frac{\lambda^j}{j!} e^{-\lambda}, j = 0, 1, 2 \ldots$$

The mean and variance of the Poisson distribution are $E(N) = Var(N) = \lambda$. (Note: $\sum_{j=0}^{\infty} P(N = j) = e^{-\lambda} \sum_{j=0}^{\infty} \frac{\lambda^j}{j!} = 1$ since $\sum_{j=0}^{\infty} \frac{\lambda^j}{j!} = 1 + \lambda + \frac{\lambda^2}{2!} + \frac{\lambda^3}{3!} + \cdots = e^{\lambda}$)

Comp. Example 3.1: estimate the probability there are no more than 2 *Eco*RI sites in a DNA molecule of length 10,000 assuming equal base frequencies and $p = 0.00024$. Here, $\lambda = 2.4$, and

$$P(N \leq 2) \cong \frac{\lambda^0}{0!} e^{-\lambda} + \frac{\lambda^1}{1!} e^{-\lambda} + \frac{\lambda^2}{2!} e^{-\lambda} = 0.569709$$

Can be obtained in R using `ppois(2,2.4)`. "In other words, more than half the time, molecules of length 10,000 and uniform base frequencies will be cut by *Eco*RI two times or less."

3.2.5. The Poisson Process: translate n into length (l) and p into rate (μ). The mean is thus $length \times rate$ (μl).

$$P(k \text{ events in } (x, x + l)) = \frac{(\mu l)^k}{k!} e^{-\mu l}, k = 0, 1, 2, \ldots$$

Moreover, provided we have two non-overlapping intervals with

$x < x + l_1 \leq y < y + l_2$, then the lengths of the intervals simply add:

$$P(k\ events\ in\ (x, x + l_1) \cup (y, y + l_2)) = \frac{(\mu(l_1 + l_2))^k}{k!} e^{-\mu(l_1+l_2)}, k = 0, 1, 2, \ldots$$

3.3 Continuous Random Variables: RVs such as the Binomial and Poisson are called discrete since the RV takes on (at most countably infinite) number of points; here the RV takes values in an interval of whole real line. Further, probability formula are called probability density functions ("pdf's", denote $f(x)$), and sums are replaced by integrals.

We already saw the uniform distribution on (a, b):

$$f(x) = \begin{cases} \frac{1}{b-a}, \text{for } a < x \leq b \\ 0, \text{otherwise} \end{cases}$$

Here, the mean and variance are $EX = \frac{a+b}{2}$ and $VarX = \frac{(b-a)^2}{12}$

The exponential distribution with parameter λ has pdf:

$$f(x) = \begin{cases} \lambda e^{-\lambda x}, \text{for } x \geq 0 \\ 0, \text{otherwise} \end{cases}$$

Here, the mean and variance are $EX = \frac{1}{\lambda}$ and $VarX = \frac{1}{\lambda^2}$

The standard normal distribution has pdf:

$$\phi(z) = \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}z^2}, -\infty < z < \infty$$

Here, the mean and variance are $EZ = 0$ and $VarZ = 1$, and we write $Z \sim N(0,1)$. Note if $Z \sim N(0,1)$ and $X = \mu + \sigma Z$, then $X \sim N(\mu, \sigma^2)$; equivalently, $Z = \frac{X-\mu}{\sigma} \sim N(0,1)$.

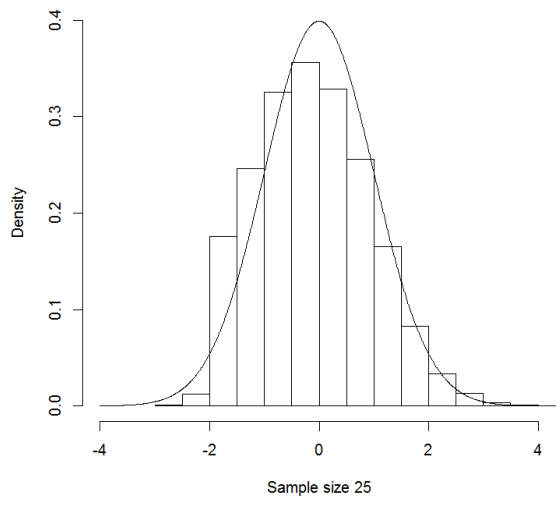
We can easily use R to find Normal probabilities:

$$X \sim N(1,4): P(X \leq 1.5) = P(Z \leq 0.25) = 0.5987$$

> pnorm(1.5,1,2) → gives [1] 0.5987063

3.4 The Central Limit Theorem (CLT): applies to averages and sums: $X_1, X_2, \ldots, X_n$ are iid with mean μ and variance σ^2, then the average is $\overline{X}_n = \frac{1}{n}(X_1 + \cdots + X_n)$. Then, the expected value and variance of $\overline{X}_n$ are μ and $\frac{\sigma^2}{n}$, so that $\frac{\overline{X}_n - \mu}{\sigma/\sqrt{n}}$ has expected value 0 and variance 1. The CLT states that for n sufficiently large, $\frac{\overline{X}_n - \mu}{\sigma/\sqrt{n}}$ behaves like a Standard Normal RV. This applies too to sums since $\frac{\overline{X}_n - \mu}{\sigma/\sqrt{n}} = \frac{\sum_{i=1}^{n} X_i - n\mu}{\sigma\sqrt{n}}$. Next, we use this to approximate the Binomial distribution, as done in STAT-335.

```
bin25<-rbinom(10000,25,0.25)
mean<-25*0.25
sd<-sqrt(25*0.25*0.75)
bin25<-(bin25-mean)/sd
hist(bin25,xlim=c(-4,4),ylim=c(0,0.4),
prob=T,xlab="Sample size 25",main="")
x<-seq(-4,4,length=1000)
lines(x,dnorm(x))
```



Next, continuing the previous Q: in a random uniformly distributed DNA sequence of length 1000, find the probability of at least 280 A's. $N \sim Binomial(n = 1000, p = 0.25)$, so we know that $E(N) = 1000 \times 0.25 = 250$ and $sd(N) = \sqrt{1000 \times 0.25 \times 0.75} = 13.6931$; using the Normal Approximation (without the continuity correction), we get

$$P(N \geq 280) \cong P\left(Z \geq \frac{280 - 250}{13.6931}\right) = P(Z \geq 2.19) = 0.014$$

1-pnorm((280-250)/sqrt(187.5)) → gives [1] 0.01422987

3.4.1 Confidence Interval for Binomial Proportion:

For sufficiently large n, since $\frac{\hat{p}-p}{\sqrt{\hat{p}(1-\hat{p})/n}}$ is approximately Standard Normal, we can find a 95% CI for p by using the interval:

$$\left(\hat{p} - 1.96\sqrt{\frac{\hat{p}(1-\hat{p})}{n}}, \hat{p} + 1.96\sqrt{\frac{\hat{p}(1-\hat{p})}{n}}\right)$$

For 90%, use 1.645 in place of 1.96. See Homework Ex.10 on p.98.

3.4.2 Maximum Likelihood Estimation:

This is a technique to estimate unknown parameters such as p above: we form the "likelihood" of the parameter given the data, and use basic calculus to maximize it; this gives the maximum likelihood estimate (MLE). To illustrate let $X_1, X_2, \ldots, X_n$ be n iid Bernoulli RVs with success probability p; then each probability mass function is $f(x) = p^x(1-p)^{1-x}$. By independence, the joint probability function is (with $k = \sum_{i=1}^{n} x_i$)

$$f(x_1, x_2, \dots, x_n) = p^{\sum_{i=1}^{n} x_i}(1-p)^{n-\sum_{i=1}^{n} x_i} = p^k(1-p)^{n-k}$$

(This is the same as the likelihood for the binomial distribution but with the $\binom{n}{k}$ term dropped since it is not a function of p.)

We now view this joint probability function as the likelihood function of p and write:

$$L(p) = p^k(1-p)^{n-k}$$

We wish to find the value of p which maximizes $L(p)$; but this is also the value of p which maximizes $log\big(L(p)\big) = l(p) = k \times log(p) + (n-k) \times log(1-p)$. Here, we set $\frac{k}{p} - \frac{n-k}{1-p} = 0$. This gives the MLE, $\hat{p}_{MLE} = \hat{p} = \frac{k}{n}$. See also Homework #11, p.98.

3.5 Restriction Fragment Length Distributions:

Suppose N (number of restriction sites) follows a Poisson process with rate λ per bp; then probability of observing k sites in an interval of length l bp is $P(N = k) = \frac{(\lambda l)^k}{k!} e^{-\lambda l}, k = 0, 1, 2, \dots$
Also, if there is a site at y, probability that a restriction fragment length X is larger than x is

$$P(X > x) = P\big(no\ events\ in\ (y, y+x)\big) = e^{-\lambda x}$$

So, $P(X \le x) = \int_0^x f(u)du = 1 - e^{-\lambda x}$, and the density for X is (by differentiation) $f(x) = \lambda e^{-\lambda x}$. So, if the number of sites (N) follows the Poisson distribution, then the distance between the sites (X) has an exponential (λ) distribution; mean length = $1/\lambda$.

3.5.1. Application to Data: digest bacteriophage lambda with *Alu*I; then the actual restriction fragment lengths and histogram are:

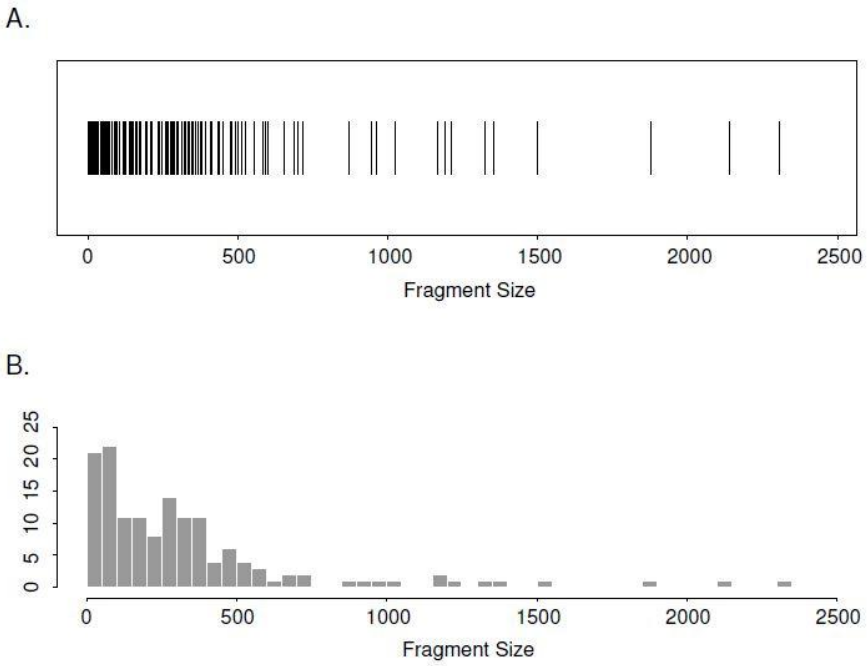


Fig. 3.3. Fragments produced by *Alu*I digestion of bacteriophage lambda DNA. Panel A: Lengths of individual fragments. Panel B: Histogram of fragment sizes.

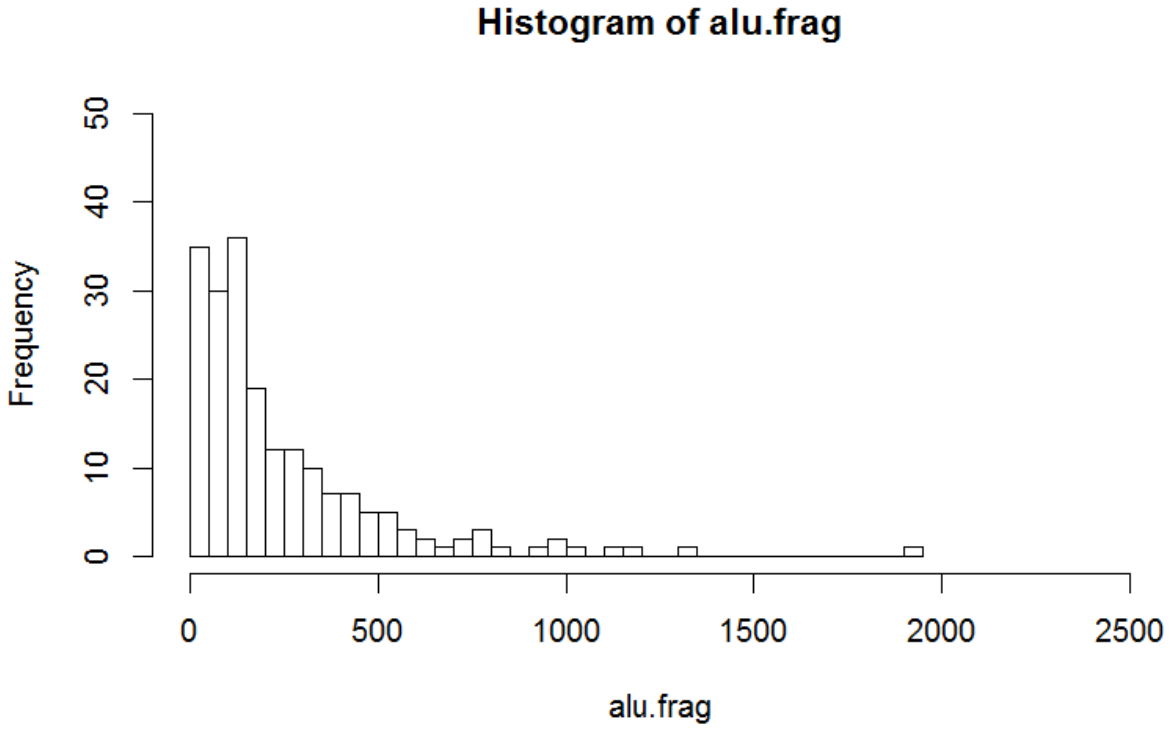
Here, $\lambda = 48502 \times 0.003906 = 189.46$ and $x = \frac{1000}{48502}$, so $x\lambda = 3.906$, so the probability of a fragment being longer than 1000 bp is $e^{-3.906} = 0.0201$. Since we observed (see Table 3.1 on p.73) 143 such fragments, we would expect to see $143 \times 0.0201 = 2.88$ or about 3 longer than 1000 bp. Since there are actually 10 observed, "[t]here is some evidence that our simple probability model does not describe the longer fragments very well." Hence, we now turn to a simulation.

3.5.2. Simulating Restriction Fragment Lengths:

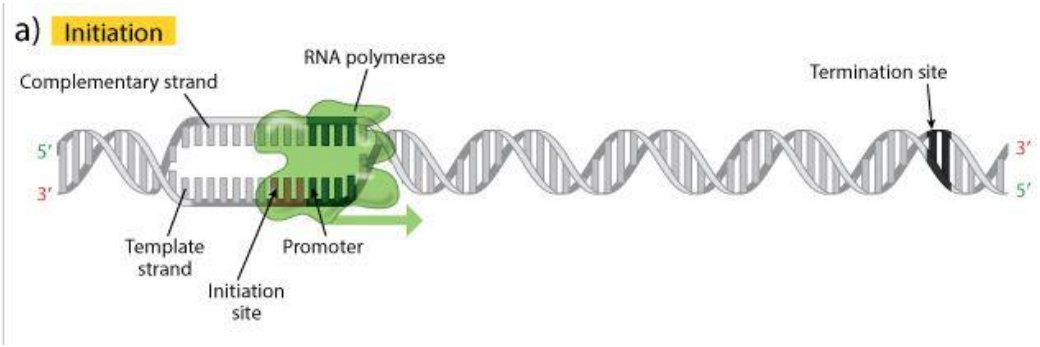
The above Poisson/exponential model "misses" somewhat, so let's now simulate using the *iid* model; see pp. 85-7.

```
  # inseq: vector containing input DNA sequence,
  # A=1, C=2, G=3, and T =4
  # seq: vector for restriction site, length m, coded as above.
  # Make/initialize vector to hold site
  # positions found in inseq

  rsite <- function(inseq, seq){
    xxx <- rep(0,length(inseq))
    m <- length(seq)   # Length of restriction recognition sequence
    truth <- rep(0, m) # To record whether pos. of inseq match seq
      # Check each position to see if a site starts there.
    for(i in 1:(length(inseq) - (length(seq) - 1))) {
      for(j in 1:m) {
        if(inseq[i + j - 1] == seq[j]) {
          truth[j] <- 1 # Record match to jth pos.
        }
      }
      if(sum(truth[]) == m){ # Check whether all positions match
        xxx[i] <- i # Record site if all positions match
      }
      truth <- rep(0, m) # Reinitialize for next loop cycle
    }
    # Write vector of restriction site positions stored in xxx.
    L <- xxx[xxx > 0]
    return(L)
  }
x<-c(1:4)
propn<-rep(0.25,4)
seq2<-sample(x,48500,replace=T,prob=propn)
alu1<-c(1,3,2,4)
alu.map<-rsite(seq2,alu1)
length(alu.map)
alu.map[1:10]

flengthr<-function(rmap,N) {
  frags<-rep(0,length(rmap))
  rmap<-c(rmap,N)
  for(i in 1:(length(rmap)-1)) {
    frags[i]<-rmap[i+1]-rmap[i]
  }
  frags<-c(rmap[1],frags)
  return(frags)
}
alu.frag<-flengthr(alu.map,48500)
alu.frag[1:10]
length(alu.frag[alu.frag>=1000])
```

We're looking for *Alu*I, AGCT = 1324, and the simulated results (both number and lengths) match the actual results on top of p.85. We could do a more rigorous analysis of the degree of matching as per p.98 exercise 13 (χ^2 goodness of fit test).



3.6 k-word Occurrences: we wish to discover functional sites in DNA, and we use promoter sequences: "promoters are gene regions where RNA polymerase binds to initiate transcription. We wish to find k-words that distinguish promoter sequences from average genomic sequences. Because promoters are related by function, we expect to observe k-words that are over-represented within the promoter set compared with a suitable null set. These k-words help to identify DNA 'signals' required for promoter function." Images below taken from:
http://www.nature.com/scitable/topicpage/dna-transcription-426#



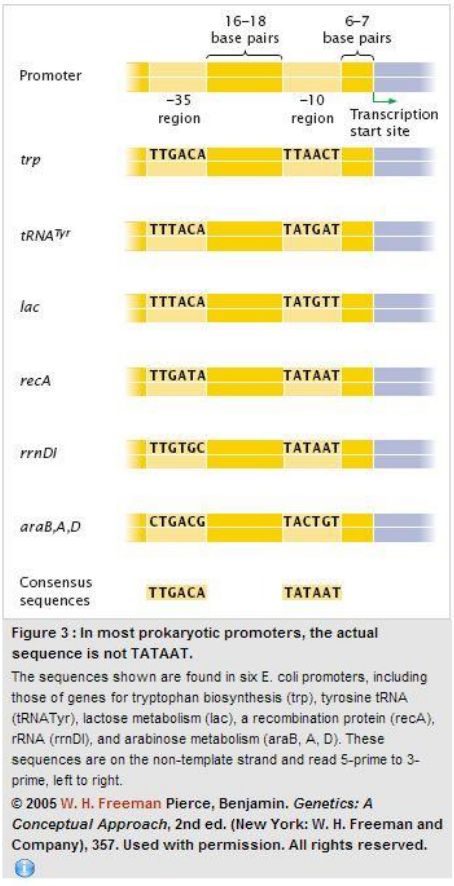


Figure 3 : In most prokaryotic promoters, the actual sequence is not TATAAT.

The sequences shown are found in six E. coli promoters, including those of genes for tryptophan biosynthesis (trp), tyrosine tRNA (tRNATyr), lactose metabolism (lac), a recombination protein (recA), rRNA (rrnDI), and arabinose metabolism (araB, A, D). These sequences are on the non-template strand and read 5-prime to 3-prime, left to right.

Consider pp.509-511: $N = 49$ promoter sequences of length $L = 51$ from *E.coli*. "We ask whether there are an unusual number of occurrences of each word in the promoter regions." In this case, the authors use $k = 4$. Under this *iid* model, "expect4.ec" in Comp. Ex. 3.6 calculates the expected frequencies for each of the $4^4 = 256$ words in 49 promoters of length 51; these range from $49 \times 48 \times 0.246^4 = 8.6134$ to $49 \times 48 \times 0.254^4 = 9.7898$.

Our first approach for word counting is to count the number of occurrences of each word ("tcount" on pp.91-2), and to compare as in Table 3.2 on p.92 the observed frequencies with the expected frequencies. Words that seem to be overexpressed are `TTTT`, `CATT`, `AATT`, `TAAT`, … (words with lots of `A`'s and `T`'s); but are these results "statistically significant"?

But there is a sticky issue here with whether or not we count overlaps, so we drop this idea, and proceed as in the following…

Our second approach for word counting here is to count the number of promoter sequences (N_w) in which the given word occurs at least once (since only one occurrence is sufficient for function). So, since a word appears at least once in a promoter sequence or not, we'll use the Normal approximation to the Binomial, with simulation to approximate the respective $p_w = P(w\ occurs\ at\ least\ once\ in\ a\ 51-letter\ sequence)$

```
ec.prom<-matrix(scan("f:aaa Teaching/a2 STAT337_437 Quant
Bioinformatics/C Spring 2014/Ec.table3.txt"),nrow=49,byrow=T)
prom.ncount<-Ncount4(ec.prom,4)
prom.ncount[1,1,1,1]
[1] 13
ec.sim<-matrix(nrow=5000,ncol=51)
for(i in 1:5000){
  ec.sim[i,]<-sample(x,51,replace=T,prob.ec)
}
sim.count<-Ncount4(ec.sim,4)
sim.count[1,1,1,1]/5000
[1] 0.1294
```

Calculated statistics ($Z_w = \frac{N_w - 49p_w}{\sqrt{49p_w(1-p_w)}}$) and p-values are on p.95; note that due to the multiple testing issue, we use the cut-off of $\alpha = 0.001$. These methods are naïve and $k = 4$ may well be too small, but these techniques have "allowed researchers to make the initial pattern discovery."