COMP 5970/6970-004
Computational Biology: Genomics and Transcriptomics
Lecture notes 2: 1/18/2022

Haynes Heaton

Spring, 2022

Lecture Objectives

- Review global alignment
- Break down Bayes' theorem and its application
  - Posterior probability
  - Likelihood
  - Prior probability
  - Total marginal likelihood
- Distributions
  - *Bernoulli*
  - Geometric
- Maximum Likelihood Estimation (MLE)

1 Global alignment review

Refer to Lecture notes 1 Global Alignment and Backtrace/Viterbi Algorithm sections.

Bayes' theorem

Bayes' rule has the following form.

$$p(A|B) = \frac{p(B|A)p(A)}{p(B)} \tag{1}$$

But this is not very intuitive and difficult to grasp. It is generally used when we have multiple different hypotheses or models of the data which may be true. And we have some data which we can look at to determine which hypothesis or model fits the data better and by how much. So using M for model and D for data, lets rewrite this as

$$p(M|D) = \frac{p(D|M)p(M)}{p(D)} \tag{2}$$

Still this is confusing. Let's expand the denominator and assume we have two competing hypotheses which follow two models M_1 and M_2

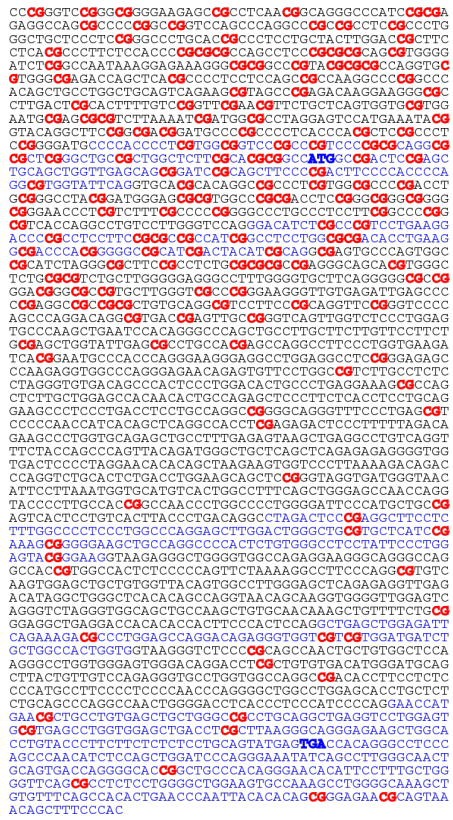
$$p(M_1|D) = \frac{p(D|M_1)p(M_1)}{p(D|M_1)p(M_1) + p(D|M_2)p(M_2)} \tag{3}$$

Now this is beginning to take form. Ignoring $p(M_1)$ and $p(M_2)$, or if these values are equal (and thus cancel out), Bayes' rule is just normalizing $p(D|M_1)$ by the total. Lets break down what each of these terms are called and what they mean.

- $p(M_1|D)$ is the **posterior probability** and is the value of interest we wish to compute. It is probability of hypothesis 1.
- $p(D|M_1)$ is the **likelihood**. This is computed from a statistical model. The simplest models are probability density functions of distributions. The simplest distribution is the *Bernoulli* distribution which is just a binary yes/no coin flip with a given probability.
- $p(M_1)$ is the **prior probability** of the hypothesis. This may be known, estimated, guessed, or treated as equal among the options depending on the problem.
- $p(D|M_1)p(M_1) + p(D|M_2)p(M_2)$ is the **total marginal likelihood**. This is the normalization factor

1.1 Example: CpG Islands

In vertebrate genomes, C's followed by G's occur at a much lower rate than expected by random chance. At 21% G/C content in the human genome, we could expect $.21^2 = 4.41\%$. But the actual frequency is $< 1\%$. Except in certain regions, the rate of this dinucleotide pattern is much higher than expected from the rate of Gs and Cs in the overall genome. These so called CpG islands occur at the upstream portion of many genes in a region known as the promoter.



So given a section of DNA, you could assess whether this is more likely a CpG island vs background genomic DNA. Say we know that the CG rate in a CpG island is

3% and the CG rate in background genomic regions is 1%. I give you a 1000bp section of DNA with 50 CG's. Finding the likelihood of this happening given the sequence is a CpG island is easy enough. For the purpose of this example we will ignore the fact that adjacent dinucleotide sequences overlap by 1 base and you cannot have a CG directly followed by a CG because the next dinucleotide sequence starts with G. The rate is low enough that this can be ignored without a large discrepancy in the outcome.

$$p(D|\text{CpG island}) = \binom{1000}{50} p^{50}(1-p)^{950} \tag{4}$$

The naive solution here would be $p^{50}(1-p)^{950}$, but this ignores all of the different orderings in which we could see these CG occurrences. We will see that in this case, this constant will cancel out in our bayes equation.

Now there are the priors to consider. Let us make a quick approximation for a ballpark number. If our posterior is not very confident, we may want to come back to this. The size of the exome (all genes) is about $\frac{1}{50}th$ of the genome. Promoters are usually smaller than genes and most contain CpG islands, so let's say the chance of any piece of DNA being a promoter is $p(\text{CpG island}) = 0.01$ which gives $p(\neg\text{CpG island}) = 0.99$. Let us call the rate of CG's in background genomic sequence $q = 0.01$. Now we can calculate the full posterior.

$$p(\text{CpG island}|D) = \frac{\binom{1000}{50} p^{50}(1-p)^{950} p(\text{CpG island})}{\binom{1000}{50} p^{50}(1-p)^{950} p(\text{CpG island}) + \binom{1000}{50} q^{50}(1-q)^{950} p(\neg\text{CpG island})} \tag{5}$$

As you can see, the $\binom{1000}{50}$ cancels out. So we have

$$p(\text{CpG island}|D) = \frac{p^{50}(1-p)^{950} p(\text{CpG island})}{p^{50}(1-p)^{950} p(\text{CpG island}) + q^{50}(1-q)^{950} p(\neg\text{CpG island})} \tag{6}$$

Generally with enough data, we will need to calculate this in log space as the numbers we get will be so close to 0 as to be numerically unstable in the floating point representation in computers. Log scaling will

$$\begin{aligned} log(p(\text{CpG island}|D)) =& 50log(p) + 950log(1-p) + log(p(\text{CpG island})) - \\ & logsumexp(50log(p) + 950log(1-p) + log(p(\text{CpG island})), \\ & 50log(q) + 950log(1-q) + log(p(\neg\text{CpG island}))) \end{aligned} \tag{7}$$

Here we see the log transformation. The only interesting aspect of this is the log sum exp function. This is used to compute

$$logsumexp(x,y) = log(exp(log(x)) + exp(log(y))) \tag{8}$$

because we cannot compute probability additions in log space. This is computed in a numerically stable way by programming language packages such as scipy and numpy. If you need this function in a language that does not have it, look up how to implement it in a numerically stable way.

To compute this log posterior in R we need to load the package containing the logsumexp function.

```
if(!require(matrixStats)) {
    install.packages("matrixStats", repos = "http://cran.us.r-project.org")
    library(matrixStats)
}

## Loading required package:  matrixStats
```

```r
p = 0.03
q = 0.01
n = 1000
k = 50
prior = 0.01

log_posterior = k*log(p)+(n-k)*log(1-p) + log(prior) -
    logSumExp(c( k*log(p)+(n-k)*log(1-p) + log(prior),  k*log(q)+(n-k)*log(1-q) + log(1-prior)))
log_posterior
```

```
## [1] -2.842171e-14
```

```r
posterior = exp(log_posterior)
posterior
```

```
## [1] 1
```

As you can see, this calculation dramatically favors the hypothesis that this DNA sequence is a CpG island over it arising from background genomic sequence.

2 Distributions

Physical processes with a random nature give rise to distributions. If you understand which random processes produce which distributions, you can have an expectation of what different data should look like. If the data does not follow these expectations, you know that there is something else going on and you can try to figure out what is causing this discrepancy.

2.1 *Bernoulli*

The simplest distribution is the *Bernoulli* distribution. This is simply the coin flip distribution which gives a 1 if there is a positive event with probability p, and 0 otherwise.

$$\begin{cases} p(X = 1) = p \\ p(X = 0) = 1 - p \end{cases} \tag{9}$$

This is a special case of the binomial distribution which we will go over later in detail, but was the distribution used in the CpG island example above.

2.2 Geometric distribution

Prokaryotic cells like backteria have contiguous gene sequences. This means that there are no gaps in the DNA sequence that is turned into RNA and then protein between the start of the gene and the end of the gene.

DNA $\rightarrow$ RNA $\rightarrow$ protein

The **central dogma of molecular biology** is that in general, information passes from DNA to RNA to proteins and proteins do the majority of the work of the cell. DNA, as mentioned before is a string on the alphabet A,C,G,T. RNA is a string on the alphabet A,C,G,U where the DNA T is equivalent to the RNA U. Proteins, however, are made up of 20 different amino acids. Because you cannot code for 20 different amino acids with a single 4-base letter, the cell uses 3 DNA/RNA bases to code for a single amino acids. These 3 base sequences which code for different amino acids are termed **codons**. Because there are $4^3 = 64$ different 3-letter codons, there can be multiple codons that code for the same amino acid. And because

there are 64 codons, the chances of seeing any given codon is $\frac{1}{64}$. There is one start codon which initiates the **transcription** of DNA to RNA and there are 3 stop codons that halt the transcription of DNA to RNA. So given that I see a stop codon, what is the likelihood of a given distance to the first stop codon. Well, in random sequence the chances of seeing a stop codon are $p = \frac{3}{64}$. So seeing a stop codon as the very next codon after the start codon is p. The likelihood of seeing k non-stop codons and then seeing a stop codon is very simply

$$(1-p)^k p \tag{10}$$

This equation is the probability density function of the **geometric distribution** and is the distribution of lengths between some positive event given a constant probability of that event at every discrete time point.

So say I give you a piece of DNA that has 300 non-stop codons between a start codon and a stop codon. How can we tell if this is surprising or not? If we had a likelihood function for how many codons between a start and stop codon there are in real genes, we could use bayes rule. In practice, you might you an empirical distribution. An empirical distribution is a distribution taken from real data. So in this case we would look at a database of gene lengths and assign a likelihood for each length between start and stop codon according to its frequency in the dataset. Another way we could proceed is ask the question "how likely would it be for us to see k or more codons between a start and stop codon" under the assumption of random sequence (or given some observed sequence frequencies in some large non-gene sequence dataset). In this case, we could use the cumulative distribution to compute this. In R, this would be

```
pgeom(300, 3/64, lower.tail=F)

## [1] 5.297954e-07
```

As an aside, most discrete distributions have continuous variations. The exponential distribution is the continuous version of the geometric distribution, and it's probability density function is

$$\lambda e^{-\lambda x} \tag{11}$$

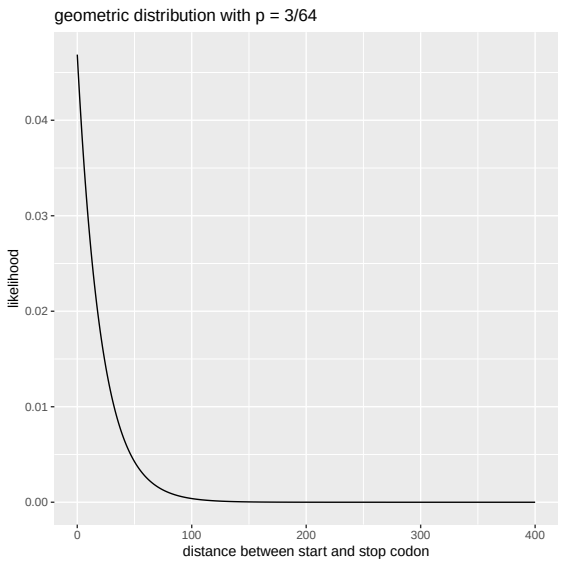
in which λ is the rate at which events occur and x is the time between events.

To get a sense of the geometric distribution, think about what value of k maximizes the likelihood. And then consider the plot of the geometric distribution with $p = \frac{3}{64}$.

```
if(!require(ggplot2)) {
    install.packages("ggplot2", repos = "http://cran.us.r-project.org")
    library(ggplot2)
}

## Loading required package:  ggplot2

p = 3/64
x = seq(0, 400)
y = (1-p)^x * p
ggplot(data.frame(x=x,y=y))+geom_line(aes(x=x,y=y))+
  ylab("likelihood")+xlab("distance between start and stop codon")+
  ggtitle("geometric distribution with p = 3/64")
```



3 Maximum Likelihood Estimation

When the distribution is known because the random process is known, but the parameters of the distribution are not known, we must estimate or infer them. One such method is choosing the parameters that maximize the likelihood of the data, or using the **maximum likelihood estimation** or MLE. In some cases this is easy, such as taking the mean of data points to find the MLE of the μ parameter of a normal distribution, or taking $\frac{1}{\text{mean length}}$ of a geometrically distributed variable to find the p value of the underlying geometric distribution. Other times it is more difficult and closed form solutions are not possible. In these cases, numerical optimization such as expectation maximization or gradient descent and combinatorial optimization methods such as Markov Chain Monte Carlo (MCMC) methods can be used to attempt to find the optimal values of the unknown parameters to these distributions.