Computational Genomics Midterm
02-510/02-710
Spring 2022

This exam has 7 questions, for a total of 100 points.

Name: ______________________________

Instructions:

- Write clearly. Unless stated otherwise, make sure to write down your steps for the calculations/derivations, do NOT just write a number.
- This exam is open book, open notes.
- You have 1 hour and 35 minutes to finish this exam.
- **DO NOT DISCUSS** the materials on this midterm (on Piazza or otherwise) until after the answer key for the midterm has been posted.
- Good luck!

No.	Topic	Max. Score	Your Score
1	Traditional Alignment	14	
2	Suffix Tries	10	
3	RNA-seq Normalization	18	
4	DE Analysis	15	
5	Motif Discovery	10	
6	Single-cell	15	
7	Population Genetics	18	

1. **[14 points] Traditional Alignment**

(a) Let $A = a_1, a_2, \ldots, a_N$, $B = b_1, b_2, \ldots, b_N$, and $C = c_1, c_2, \ldots, c_N$ be three sequences of length N. The *longest common subsequence* is the longest sequence $S = s_1, \ldots, s_K$ such that S is a subsequence of A, B, and C.

Note that a *subsequence* is defined as any sequence that can be formed by deleting one or more characters from a sequence. For example, "abd" and "be" are valid subsequences of the sequence "abcde".

Provide a dynamic programming algorithm that, given sequences A, B, and C, finds the **length** of their longest common subsequence. Briefly explain why your algorithm is correct and provide the runtime.

Solution

Initialize $Q(i,j,k) = 0$ if $i = 0$ or $j = 0$ or $k = 0$. This is because if at least one string is empty, there cannot be a common subsequence. The recurrence relation for dynamic programming is:

$$\begin{gathered} Q(i,j,k) = Q(i-1,j-1,k-1) + 1 \ \ (\text{if } a_i = b_j = c_k) \\ Q(i,j,k) = \max\{Q(i-1,j,k), Q(i,j-1,k), Q(i,j,k-1)\} \\ (\text{if } a_i \neq b_j \text{ or } b_j \neq c_k \text{ or } c_k \neq a_i) \end{gathered}$$

Runtime for this algorithm is $O(N^3)$.

(b) Suppose that we use progressive alignment (as we did in HW1 Q3) with *affine gap penalty* on three sequences and observe that the alignments below lead to the same total score:

Alignment 1:	**Alignment 2:**
$Seq1$ `ATCGTT`	$Seq1$ `ATCGTT`
$Seq2$ `ATTTTT`	$Seq2$ `ATTTTT`
$Seq3$ `AT__T_`	$Seq3$ `A___TT`

where we use match score M, gap start penalty G, gap extension penalty E, and mismatch penalty Mi. Recall from HW1 that in progressive alignment, the total score is the sum of pairwise alignment scores and that gaps aligned with each other **do not** contribute to the total score.

Which of the following should be true for E?

A. $E = Mi$
B. $E = \frac{1}{2}Mi$
C. $E = G$
D. $E = \frac{1}{2}G$

Solution

Choice C. For the new alignment, the total score is:

$$\begin{aligned}(1,2) &: 4M + 2Mi \\ (1,3) &: 3M + G + 2E \\ (2,3) &: 3M + G + 2E \\ \Rightarrow \text{Total} &: 10M + 2Mi + 2G + 4E\end{aligned}$$

Setting it equal to the total score for the original alignment, we have

$$\begin{aligned}10Mi + 2Mi + 4G + 2E &= 10M + 2Mi + 2G + 4E \\ \Rightarrow E &= G\end{aligned}$$

2. **[10 points] Suffix Tries**

Given a suffix **trie** with suffix links for a string s, provide an efficient algorithm to search for the longest substring of s that appears both in the forward and reverse directions within s. For example, for the string "banamangonamana", the longest such substring is "anaman" (or, equivalently, "namana"). Provide a runtime analysis for your algorithm in terms of $|s| = n$.

Some other examples:

- "ranrna" → "anrna"
- "rantrna" → "an"
- "racecar" → "racecar"

Solution

Using suffix links, we can solve this in $O(n)$ time. Denote the reverse of a string s as s'. Then, the goal is to find the longest common substring of s and s'; this is exactly our solution. Using suffix links and recording the maximum length of the substring of s' that we match, we can find the solution in $O(n)$ time.

Other approaches can solve this problem in $O(n^2)$ time. For example, let's introduce a new characters into our alphabet: $\$_{\text{backward}}$. Encode the suffixes of $s'\$_{\text{backward}}$ into the original suffix trie. Next, using DFS, augment each trie node with the counts of the number of \$ and $\$_{\text{backward}}$ that are leaves of the subtrie rooted at that node, and then DFS for the deepest internal node that has at least one \$ and one $\$_{\text{backward}}$ as children (this part is similar to the solution for HW2 problem 4c); this is exactly the longest common substring between s and s'. The runtime of encoding the reverse string and running DFS is $O(n^2)$; thus, the overall runtime is thus $O(n^2)$.

Notes:

- -10 for incorrect algorithm

If assumed that suffix trie given:

- -3 for not using suffix links/ greater than $O(n)$ solution
- -5 for incorrect runtime analysis

If assumed that suffix trie not given:

- -3 for brute force algorithm/ greater than $O(n^2)$ runtime
- -5 for incorrect runtime analysis

3. **[18 points] RNA-seq Normalization**

We are interested in normalizing single cell read data. We obtained single cell expression for 1000 cells. To deal with dropouts we perform the following analysis prior to using RPKM. For each *gene* we perform clustering using GMM assuming Gaussian distribution and 2 clusters. The first cluster accounts for the actual expression while the second is the drop out cluster.

(a) What parameters do we need in order to specify the distribution for the GMM? Specify the model we would need to learn including all parameters. (**Hint:** There may be parameters that can be fixed.)

Solution

For each gene i, we need 1 parameter for the fraction of the two clusters for each gene (α_i) and two distributions. The first is the actual cluster and has two parameters: $\mathcal{N}(\mu_i^1, \sigma_i^1)$. The second is for the 'dropout' cluster. Here we need to fix the mean to 0 and so only one parameter: $\mathcal{N}(0, \sigma_i^2)$.

(b) What are the problems with using a Gaussian model for the clustering (list two)?

Solution

Two problems. First, it's not appropriate for dealing with discrete count data. Second, it's not appropriate for dealing with nonnegative data.

(c) We would like to only use the 'actual' expression value for the gene when computing the RPKM analysis. For gene i in sample j, let g_i^j be the expression of the gene, α_i^j be the fraction of the gene assigned to the actual cluster and β_i^j be the fraction assigned to the dropout cluster. How can we do this? Write down the equation for the RPKM value computed for gene i in sample j.

Solution

Instead of summing all gene values and dividing by the total number of counts for each cell (M) we just sum the 'actual' part and divide by the sum of that part. In other words define:

$$M^j = \sum_i \alpha_i^j$$

Then for every gene i, we compute the following:

$$RPKM_i^j = \frac{\alpha_i^j R_i^j 10^3 10^6}{L_i M^j}$$

4. [**15 points**] **DE Analysis**

(a) Assume we obtained expression data from 10 individuals. 6 were diagnosed with lung cancer while the other 4 were healthy. What is the minimum p-value we can obtain for a specific gene if using a permutation-based method for computing the p-value (no need for exact value, a formula is fine)?

Solution

$\frac{1}{\binom{10}{6}}$

(b) Assume we have a method for computing significance for a single gene. The minimum value the method can provide is 0.00005. We performed an analysis of 2000 genes. If we determined that 50 of them are differentially expressed then:

i. If we used the most strict p-value we could, what is the corrected p-value we used according to Bonferroni?

Solution

0.1

ii. What is the FDR?

Solution

0.2%

5. **[10 points] Motif Discovery**

A motif M can be defined with a Position Weight Matrix where each element $M(i,j)$ represents the probability of having a nucleotide i at position j. For example, in the PWM shown in Table 1, $M(A,3) = 0.3$ indicates that the probability of "A" at position 3 is 0.3. b_i represents the background probability for nucleotide i, where $i \in \{A, C, T, G\}$. Now, given this PWM, answer the following questions:

	1	2	3	4
A	0.4	0.3	0.3	0.6
C	0.1	0.2	0.15	0.05
T	0.4	0.3	0.15	0.1
G	0.1	0.2	0.4	0.25

Table 1: PWM for motif M

Given the PWM in Table 1 and assuming a background probability of a for A, T and a background distribution of $0.5 - a$ for G, C at all positions. For which values of a is A more likely than B? Give your answer as a range.

A. ACAT

B. TTGC

Solution

$0 < a < 0.25$ We have $\log_2(\frac{0.4}{a}) + \log_2(\frac{0.2}{0.5-a}) + \log_2(\frac{0.3}{a}) + \log_2(\frac{0.1}{a}) > \log_2(\frac{0.4}{a}) + \log_2(\frac{0.3}{a}) + \log_2(\frac{0.4}{0.5-a}) + \log_2(\frac{0.05}{0.5-a})$. This gives $\frac{1}{a} > \frac{1}{0.5-a}$. Hence, $0 < a < 0.25$

6. [**15 points**] **Single-Cell**

(a) Consider the setting where we measure the expression of n genes across p experimental trials and wish to identify meaningful clusters of genes for downstream analysis. Note that we can treat the p different measurements for each gene as features representing the gene. In other words, we can treat the given expression data as n data points, each with p features.

Suppose that we want to use hierarchical clustering based on Euclidean distance but wish to apply some form of transformation to the data beforehand. Which of the following transformations could lead to a **different** clustering result in comparison to applying hierarchical clustering **without** any transformation? *Select **all** such choices.*

A. Scaling (i.e. multiplying each feature by some factor)

B. Rotation

C. PCA

D. t-SNE

Solution

Choices A,C,D. Only rotations are guaranteed to preserve distances between points. **Correction:** Original intention was that scaling factors can be different for different features, in which case the ordering of the distances can be altered and lead to different clustering results. However, it appears by hindsight that the specification in parentheses can be ambiguous. Scoring will just be based on whether C,D were chosen and B wasn't.

(b) We are looking at performing clustering for single-cell data. We first reduce the dimension of the data and then cluster in the lower dimension.

i. Assume we expect to have several clusters, each with very similar expression pattern for cells of the same type though with possibly different sizes. Which of the following methods is the best for performing the dimensionality reduction in such case?

A. PCA

B. SNE with fixed variance for all cells

C. MDS

D. All will perform equally well

Solution

MDS. We need a method that places major emphasis on a small distances and does not care about larger ones.

ii. Assume that instead of regular clustering we would like to perform hierarchical clustering so that we do not just obtain the clusters (cell types) themselves but also can determine relationships between cells types and group cell types into larger construct. Which of the following methods is the best for performing the dimensionality reduction in such case?

A. PCA

B. SNE with fixed variance for all cells

C. MDS

D. All will perform equally well

Solution

PCA. While SNE can also be used, it still places a lot of emphasis on close points and so can distort the overall organization of the high dimensional data. In contrast, PCA is uniform and so preserves distances at all levels.

7. [**18 points**] **Population Genetics**

(a) Consider two genotype datasets, one for Africans and the other for Europeans. The African population is significantly older than the European population. How is the linkage disequilibrium pattern different between the two populations? How is the haplotype length different between the two populations?

Solution

The African population has shorter haplotype blocks than the European population, and a lower level of linkage disequilibrium.

(b) Consider using a mixture model to learn the population structure. Let $\mathbf{X} = [X_1, \ldots, X_P]$ represent the random variables for genotypes for P loci of an individual, taking values from $\{0, 1, 2\}$. Let C denote a random variable for population label ($C = 1$ for African, $C = 2$ for Asian, and $C = 3$ for Caucasian) for the given individual. Then, the mixture model for $\mathbf{X}$ is given as

$$P(\mathbf{X}) = \sum_{i=1}^{3} P(\mathbf{X}|C = i)P(C = i),$$

where $P(\mathbf{X}|C = i) = \prod_{j=1}^{P} \text{Multinoulli}(\theta^i_{j0}, \theta^i_{j1}, \theta^i_{j2})$ and $P(C) = \text{Multinoulli}(\phi^1, \phi^2, \phi^3)$, i.e., $P(C = i) = \phi^i$.

i. Assume you are given the estimated model parameters. Explain how you would do inference and assign population label to an individual with genotype data $\mathbf{x} = [x_1, \ldots, x_P]$. (**Hint:** Compute $P(C|\mathbf{X} = \mathbf{x})$, given the estimated model parameters and data $\mathbf{x}$.)

Solution

For $i = 1, 2, 3$, apply the Bayes rule as follows:

$$\begin{aligned} P(C = i|\mathbf{X} = \mathbf{x}) &= \frac{P(\mathbf{X} = \mathbf{x}|C = i)P(C = i)}{\sum_k P(\mathbf{X} = \mathbf{x}|C = k)P(C = k)} \\ &= \frac{\prod_{j=1}^{P} \theta^i_{jx_j} \phi^i}{\sum_k \prod_{j=1}^{P} \theta^k_{jx_j} \phi^k}. \end{aligned}$$

Select i with the highest $P(C = i|\mathbf{X} = \mathbf{x})$ as the population label.

ii. Assume fitting this model with two mixture components to the following two different datasets—**a:** genotype data for individuals from two recently diverged populations and **b:** genotype data for individuals from two populations diverged a long time ago. Explain how the estimated parameters would look different between Case **a** and Case **b**.

Solution

The two sets of parameters, $(\theta^1_{j0}, \theta^1_{j1}, \theta^1_{j2})$'s for population 1 and $(\theta^2_{j0}, \theta^2_{j1}, \theta^2_{j2})$'s for population 2, represent allele frequencies in each population. In **a**, these two sets of parameters will be more similar than in **b**.