IITM-CS6024 : Algorithmic Approaches to Computational Biology | Release Date: Sep 7

Assignment 3 | Due Date : Sep 17, 23:55

Roll No: BE16B002 | Name: Anoubhav Agarwaal

1. (5 points) [Solve or search!] Answer YES/NO to these questions on the Euclidean-distance-based clustering problems seen in class. Justify each answer very briefly (one/two sentences either with your own argument or by citing a properly-formatted reference paper/book). Each of the n datapoints is of dimension m, and we seek k clusters.

   (a) (1 point) For all $m > 1$, is k-Means problem NP-hard for general k and polynomial-time solvable for $k = 1$?

   Solution:
   Yes, k-Means is NP-hard for general k for all $m > 1$.[1]
   Yes, It is polynomial-time solvable for k = 1. The mean of the n datapoints is the optimal solution.

   (b) (1 point) For all $m > 1$, is k-Centers problem NP-hard for general k and **not** polynomial-time solvable for $k = 1$?

   Solution:
   Yes, k-Centers is NP-hard for general k for all $m > 1$.[2]
   No, It is polynomial-time solvable for k = 1 and general m.
   Naive algorithm: Compute pairwise distances for all n datapoints and track the largest pairwise distance for each datapoint. For large m this has a time complexity of $O(mn^2)$. Next, pick the datapoint having the minimum largest pairwise distance. This takes $O(n)$ time. This datapoint is the optimal 1-center.

   (c) (1 point) For $m = 1$ (i.e., all data points lie on a line), is k-Means problem polynomial-time solvable for general k?

   Solution: Yes, it is polynomial-time solvable. An exact solution to 1-D k-means clustering has been developed in [3]. It uses dynamic programming to guarantee clustering optimality in $O(n^2k)$ time.

   (d) (1 point) For $m = 1$, is k-Centers problem polynomial-time solvable for general k?

   Solution: Yes, it is polynomial-time solvable. The complexity of the following dynamic programming algorithm is $O(n^2k)$. Also, this paper [4] gives an $O(nlogn)$ time algorithm.

(e) (1 point) For $m = 1$ and $k = 1$, does the optimal solution of the k-Centers problem always cost half the distance between the two farthest datapoints?

Solution: Yes, it always costs half the distance. The line segment connecting the two farthest points will contain every other datapoint. Picking a point other than the centre of this line segment will always lead to a higher cost.

Note that polynomial-time solvable means the running time is polynomial in n, k and m, and "general k" refers to a k that is not fixed to any value (such as 1) but instead provided as part of the input.

2. (5 points) [Softening k-Means] Soft k-Means clustering algorithm can be viewed as performing inference on a probabilistic mixture model (using the EM algorithm or Newton-Raphson method). Consider a random variable X modeled as a *mixture of two Gaussians* with the following density function,

$$P(X = x \mid \mu_1, \mu_2, \sigma, \pi_1, \pi_2) = \sum_{k=1}^{2} \pi_k \frac{1}{\sigma\sqrt{2\pi}} \exp\left\{-\frac{(x-\mu_k)^2}{2\sigma^2}\right\},$$

where the two Gaussians are given the labels $k = 1$ and $k = 2$; the prior probability of the class label Z is $\{P(Z = 1) = \pi_1, P(Z = 2) = \pi_2\}$; $\{\mu_k\}$ are the means of the two Gaussians; and both have standard deviation σ. Assuming that all five parameters collectively denoted θ are known, the posterior probability of the class label Z of an observed datapoint x can be written using Bayes' rule as:

$$P(Z = 1 \mid x, \theta) = \frac{1}{1 + \exp[-(w_1 x + w_0)]}; \quad P(Z = 2 \mid x, \theta) = \frac{1}{1 + \exp[+(w_1 x + w_0)]}$$

(a) (1 point) Provide the expression for w_1 and w_0.

Solution: The posterior probability of the class label Z = 1 of an observed datapoint x can be obtained using Bayes' rule as:

$$P(Z = 1 \mid x, \theta) = \frac{P(x, \theta \mid Z = 1)P(Z = 1)}{P(x, \theta \mid Z = 1)P(Z = 1) + P(x, \theta \mid Z = 2)P(Z = 2)}$$

$$P(Z = 1 \mid x, \theta) = \frac{1}{\frac{\cancel{P(x,\theta \mid Z=1)\times\pi_1}}{\cancel{P(x,\theta \mid Z=1)\times\pi_1}} + \frac{P(x,\theta \mid Z=2)\times\pi_2}{P(x,\theta \mid Z=1)\times\pi_1}} = \frac{1}{1 + \frac{\pi_2 \times P(x,\theta \mid Z=2)}{\pi_1 \times P(x,\theta \mid Z=1)}}$$

$$P(Z = 1 \mid x, \theta) = \frac{1}{1 + \frac{\pi_2 \times \cancel{\frac{1}{\sigma\sqrt{2\pi}}} e^{-\frac{(x-u_2)^2}{2\sigma^2}}}{\pi_1 \times \cancel{\frac{1}{\sigma\sqrt{2\pi}}} e^{-\frac{(x-u_1)^2}{2\sigma^2}}}} = \frac{1}{1 + \frac{\pi_2}{\pi_1} \times e^{-\frac{(u_2^2 - u_1^2) + 2x(u_1 - u_2)}{2\sigma^2}}}$$

$$P(Z = 1 \mid x, \theta) = \frac{1}{1 + \frac{\pi_2}{\pi_1} \times e^{-\frac{(u_2^2 - u_1^2) + 2x(u_1 - u_2)}{2\sigma^2}}} = \frac{1}{1 + e^{-(w_1 x + w_0)}}$$

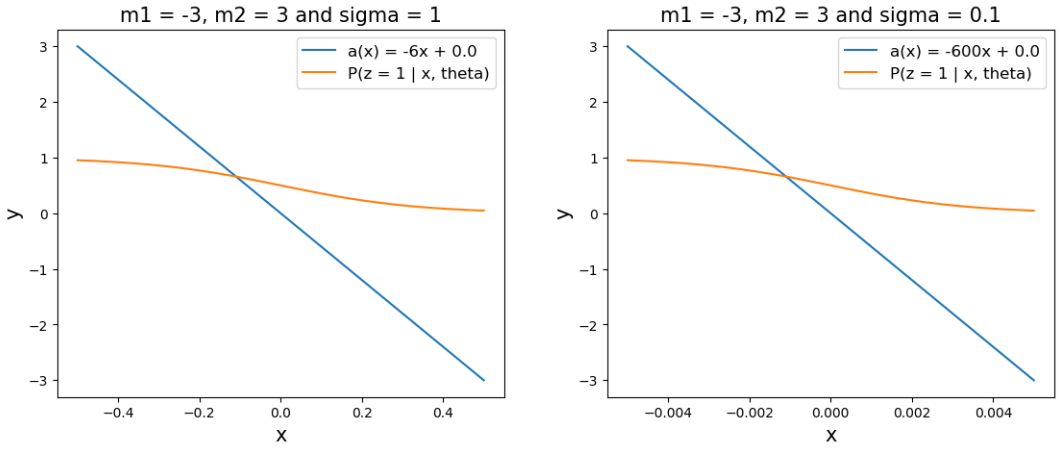
Assuming the prior probabilities of the class label Z are equal.

$\{P(Z=1)=\pi_1, P(Z=2)=\pi_2 \ \ and\ \pi_1=\pi_2=0.5\}$. On comparing the L.H.S and R.H.S in the above equation we get:

$$w_1 = \frac{u_1 - u_2}{\sigma^2}; \qquad w_0 = \frac{u_2^2 - u_1^2}{2\sigma^2}$$

(b) (1 point) Plot $a(x) = w_1 x + w_0$, and $P(z = 1 \mid x, \theta) = 1/(1 + \exp[-a(x)])$ as a function of x for the means $m_1 = -3, m_2 = 3$ and equal class prior probabilities $\pi_1 = \pi_2 = 1/2$. Show your plots for two values of σ ($\sigma = 1$ and $\sigma = 1/10$).

Solution: The plots for the two values of σ:



(c) (2 points) We would like to find the θ that maximizes the likelihood function $L(\theta; \{x_n\}_{n=1}^N) = \prod_{n=1}^N P(x_n \mid \theta)$, assuming σ is known and the two classes have equal prior probability. Derive partial derivatives of the natural log of L denoted LL wrt μ to show that the Newton-Raphson update step ($\mu' = \mu - \left[\frac{\partial LL}{\partial \mu} / \frac{\partial^2 LL}{\partial \mu^2}\right]$) to find a local maxima yields the iterative update used in soft k-means: $\mu_k' = (\sum_n P(Z = k \mid x_n, \{\mu_k\})/(\sum_n P(Z = k \mid x_n, \{\mu_k\}))$. (Hint: You can approximate second derivative as specified in Exercise 22.5 of David MacKay's book.)

Solution: Let L denote the likelihood and LL denote the log likelihood. The first and second partial derivative of the log likelihood with respect to u_k is

given by:

$$L(\theta; \{x_n\}_{n=1}^N) = \prod_{n=1}^{N} P(x_n \mid \{u_k\}, \sigma)$$

$$LL = \log L(\theta; \{x_n\}_{n=1}^N) = \sum_{n=1}^{N} \log P(x_n \mid \{u_k\}, \sigma)$$

$$LL = \sum_{n=1}^{N} \log \sum_{j=1}^{2} \pi_j \frac{1}{\sigma\sqrt{2\pi}} \exp\left\{-\frac{(x-\mu_j)^2}{2\sigma^2}\right\}$$

$$\frac{\partial LL}{\partial u_k} = \sum_{n=1}^{N} \frac{\partial}{\partial u_k}\left\{\log \sum_{j=1}^{2} \pi_j \frac{1}{\sigma\sqrt{2\pi}} \exp\left\{-\frac{(x_n-\mu_j)^2}{2\sigma^2}\right\}\right\}$$

$$\frac{\partial LL}{\partial u_k} = \sum_{n=1}^{N} \left\{\frac{\pi_k \frac{1}{\sigma\sqrt{2\pi}} \exp\left\{-\frac{(x_n-\mu_k)^2}{2\sigma^2}\right\}}{\sum_{j=1}^{2} \pi_j \frac{1}{\sigma\sqrt{2\pi}} \exp\left\{-\frac{(x_n-\mu_j)^2}{2\sigma^2}\right\}} \times \frac{\partial}{\partial u_k}\left\{-\frac{(x_n-\mu_k)^2}{2\sigma^2}\right\}\right\}$$

$$\frac{\partial LL}{\partial u_k} = \sum_{n=1}^{N} \left\{P(Z=k \mid x_n,\ \theta) \times \frac{x_n - u_k}{\sigma^2}\right\} \quad (1)$$

$$\frac{\partial^2 LL}{\partial {u_k}^2} = \sum_{n=1}^{N} \frac{\partial}{\partial u_k}\left\{P(Z=k \mid x_n,\ \theta) \times \frac{x_n - u_k}{\sigma^2}\right\}$$

$$\frac{\partial^2 LL}{\partial {u_k}^2} = \sum_{n=1}^{N} \cancel{\frac{\partial}{\partial u_k} P(Z=k \mid x_n,\ \theta)} \times \frac{x_n - u_k}{\sigma^2} + P(Z=k \mid x_n,\ \theta) \times \frac{\partial}{\partial u_k}\frac{x_n - u_k}{\sigma^2}$$

In David MacKay's book, the term $\frac{\partial}{\partial u_k} P(Z = k \mid x_n, \theta)$ is neglected for calculating the second derivative. Thus, the second derivate of the log likelihood is given by:

$$\frac{\partial^2 LL}{\partial {u_k}^2} = -\sum_{n=1}^{N} \frac{P(Z=k \mid x_n,\ \theta)}{\sigma^2}; \quad (2)$$

On substituting the first derivate (1) and second derivative (2) in the **Newton-Raphson** update step:

$$\mu_k' = \mu_k - \left[\frac{\partial LL}{\partial \mu_k} \Big/ \frac{\partial^2 LL}{\partial \mu_k^2}\right]$$

$$\mu_k' = \mu_k - \frac{\sum_{n=1}^{N} \left\{P(Z=k \mid x_n,\ \theta) \times \frac{x_n - u_k}{\cancel{\sigma^2}}\right\}}{-\sum_{n=1}^{N} \frac{P(Z=k \mid x_n,\ \theta)}{\cancel{\sigma^2}}}$$

$$\mu'_k = \mu_k + \frac{\sum_{n=1}^{N} x_n \times P(Z = k \mid x_n,\ \theta)}{\sum_{n=1}^{N} P(Z = k \mid x_n,\ \theta)} - u_k \times \frac{\sum_{n=1}^{N} P(Z = k \mid x_n,\ \theta)}{\sum_{n=1}^{N} P(Z = k \mid x_n,\ \theta)}$$

$$\mu'_k = \frac{\sum_{n=1}^{N} x_n \times P(Z = k \mid x_n,\ \theta)}{\sum_{n=1}^{N} P(Z = k \mid x_n,\ \theta)}$$

(d) (1 point) Based on answers to the last two parts, for what value of σ (1 or 1/10) does the above soft k-means algorithm behave like the hard k-means algorithm?

Solution: For $\sigma = 0.1$, the above soft k-means algorithm behaves like hard k-means. Look at the x-axis range in the plots in part (b). The posterior probability switches more rapidly from zero to one in the case of $\sigma = 0.1$.

3. (10 points) [Going far to be centered...]: Recall the FarthestFirstTraversal heuristic seen in class for solving the k-Centers clustering problem, where you choose an arbitrary datapoint as a cluster center and pick subsequent cluster centers sequentially by choosing the farthest point from the set of currently chosen clusters.

   (a) (7 points) Implement the FarthestFirstTraversal heuristic as specified in this HackerRank contest.

   (b) (2 points) We saw in class that the cost of a solution (SOLN) returned by this heuristic is guaranteed to be at most twice the cost of an optimal solution (OPT) i.e., $SOLN \leq 2\ OPT$. Are there input instances (sets of datapoints) where the solution cost can be the worst value of **exactly** twice the optimal cost i.e., $SOLN = 2\ OPT$? If so, present such worst-case input instances (for both $k = 1$ and $k > 1$).

Solution: For k = 1 and dimension m = 1, consider a set of two points at 1 unit of distance. The optimal center is at 0.5 unit distance from each point. Whereas, the FFT will select any one vertex, yielding cost of 1 unit.

For m = 1 and general k, consider a set of 2*k points spaced at increments of 1 unit of distance. The optimal centers yield a cost of 0.5. They will be present at the centers of the k adjacent pairs. Whereas, the FFT heuristic will yield a cost of 1 unit.

For m = 2 and k = 1, consider a rectangle i.e. a set of four points. The optimal center is the centroid, yielding a cost of 0.5*length of diagonal. The FFT solution will select one of the vertices. This yield a cost of the length of diagonal.

For m = 2 and k = 2, consider a rectangle i.e. a set of four points. The optimal solution is the centers of the shorter side of the rectangle, yielding a

cost of 0.5*length of the shorter side. The FFT solution will select diagonally opposite vertices yielding a cost of the length of the shorter side.

(c) (1 point) Let $SOLN$ be the cost of a solution returned by running FarthestFirst-Traversal on any input instance of the k-Centers problem as above. While showing the 2-approx. guarantee in class, we used a key fact that any two centers chosen in this solution C_i, C_j $(i \neq j)$ must be at an Euclidean distance at least $SOLN$ apart i.e., $d(C_i, C_j) \geq SOLN$. Prove this as clearly and concisely as possible.

Solution: Starting with n datapoints in a space, number all the points in it using a farthest-first traversal. For any point i, let R_i be the minimum distance of point i to the set of points $\{1, 2, ..., i-1\}$.[5]

$$R_i = d(i, \{1, 2, ..., i-1\})$$

Let C_k be the k-clustering. We need to show that its cost is given by R_{k+1} and thus SOLN $= R_{k+1}$.
First, we need to prove that $R_j \leq R_i \ \ \forall\ j > i$. Using this and SOLN $= R_{k+1}$, we can show that $d(C_i, C_j) \geq SOLN$.

$$\begin{aligned} R_j &= d(j, \{1, 2, ..., j-1\}) \\ &\leq d(j, \{1, 2, ..., i-1\}) \\ &\leq d(i, \{1, 2, ..., i-1\}) = R_i \end{aligned} \tag{1}$$

In a k-clustering, the distance from any point $m > k$ to its closest center is $d(m, \{1, 2, ..., k\}) \leq d(k+1, \{1, 2, ..., k\}) = R_{k+1}$. Hence, $SOLN = R_{k+1}$ (2)
Without loss in generality, assume $i < j$.

$$\begin{aligned} i < j < k+1 & \\ R_i \geq R_j \geq R_{k+1}; \quad & by\ (1) \\ R_j \geq SOLN \quad ; \quad & by\ (2) \\ d(j, \{1, 2, ..., i, ..., j-1\}) \geq SOLN & \\ d(j, i) \geq d(j, \{1, 2, ..., i, ..., j-1\}) & \\ d(j, i) = d(C_j, C_i) \geq SOLN & \end{aligned}$$

4. (10 points) [Dissecting a food-poisoning outbreak: Part 2] As you may remember, we assembled a campus bacterial isolate's reads using reference-free or *de novo* assembly in Assignment 1. After consulting with the researcher, we found the isolate was actually a mutant strain of Staphylococcus aureus bacteria. In this assignment, we perform a reference-based assembly by mapping the reads of the campus isolate against the reference genome of **Staphylococcus aureus** (available in **wildtype.fna**). This genome

assembly will show where the campus bacteria is different/mutant relative to the reference genome, and hence enable future studies of whether these mutations lead to severe health impact on hosts.

Steps to do using any of the Galaxy servers (Server1, Server2):

1. Get the campus isolate's reads and reference genome.
2. Read-mapping with a Burrows-Wheeler indexing method called BWA-MEM.
3. Post-processing mapped reads (includes filtering and removal of duplicates).
4. Visualize campus isolate vs. reference genome using JBrowse genome browser.

helpful link

(a) (2 points) What is a reference genome? For each organism, several possible reference genomes may be available (e.g., hg19 and hg19 for human). What do they correspond to?

Solution: Organisms of the same species have some variations at the gene level. The Reference genome is a template genome assembled by scientists. It is a representative example of a species' set of genes. It is built using the information from the DNA of several external donors and iterated based on the latest scientific findings. Once a genome has been sequenced, it is compared with the reference genome. The variations in the regions of DNA could answer important biological questions.[6], [7]

The reference genome is continuously updated based on the latest scientific findings. Hence, many versions of the reference genome are available for each organism. For example, GRCh37 and GRCh38 are human genome assemblies by the Genome Reference Consortium (GRC). GRCh38 was released four years after the GRCh37 release in 2009. The updates in GRCh37 include repairing of incorrect reads, the addition of alternate loci, etc. Also, reference genome assemblies are available from several sources and are named differently. GRCh37 (by GRC) and hg19 (by UCSC) are the same but named differently.[8]

(b) (4 points) In Step 3 above where we remove duplicates, we treat multiple identical reads that map to the same location in the reference genome as a single read. Why do you think the data contains such duplicate reads, and why is it important to de-duplicate them before "finding mutations" (aka "calling variants")?

Solution: The data contains duplicate reads because while performing Illumina sequencing, we take the genome and break it up into fragments of length 100-300 bp (through sonication). We then ligate adapters to both ends of the fragment and PCR amplify the fragments.PCR duplicates of the reads arise in this step.

We do this because, in the process of sequencing, many of the reads get destroyed. Also, there are sequencing errors. Thus, for more accurate sequencing, we usually have a genome coverage of 10-30x. Genome coverage is the average number of reads that span a nucleotide in the genome — the higher the coverage, the better the data quality.

The primary purpose of removing duplicates is to mitigate the effects of PCR amplification bias introduced during library construction (mentioned above). On mapping samples to a reference genome, duplicate reads from PCR amplification result in areas of disproportionally high coverage. Duplicates are often the cause of significant skew in allelic ratios. Sequencing errors incorporated post-amplification can affect both sequences- and coverage-based analysis methods, such as variant calling. The introduced errors can create false positive SNPs, and ChIP-Seq, where artificially inflated coverage can skew the significance of certain locations.[9]

(c) (2 points) Use JBrowse to report what allele (A, C, G or T) is found in the reference genome and mutant-bacteria genome in position 24388 in the reference genome. Report the same for a second position 84015.

Solution: Postprocessing performed on the mapped reads:

1. **Filter the paired-end reads** of all samples to retain only those read pairs, for which both the forward and the reverse read have been mapped to the reference successfully (using Filter SAM or BAM, output SAM or BAM tool).
2. **De-duplicate reads** (using RmDup*). [10]

*RmDup version 2.0 was giving an error. Thus, RmDup version 1.0 was used to remove duplicates.

At position 24388, we find allele A (in one strand and T in reverse complement strand) in the reference genome. In the mutant-bacteria genome, we find allele G.

At position 84015, we find allele A (in one strand and T in reverse complement strand) in the reference genome. In the mutant-bacteria genome, we find the same alleles as the reference genome. **No, alternate allele was found in my analysis at position 84015.**

(d) (2 points) Again use JBrowse to report how many reads support the reference vs. alternate allele at each of the above two positions. Which position do you think has a higher chance (statistical significance) of being a true mutation?

Solution:

At position 24388, there are 23 reads in total. All of them support the mutant's alternate allele G.

At position 84015, there are 14 reads in total. All of them support the reference allele.

For my analysis, due to obvious reasons, there is a higher chance for position 24388 to be a true mutation.

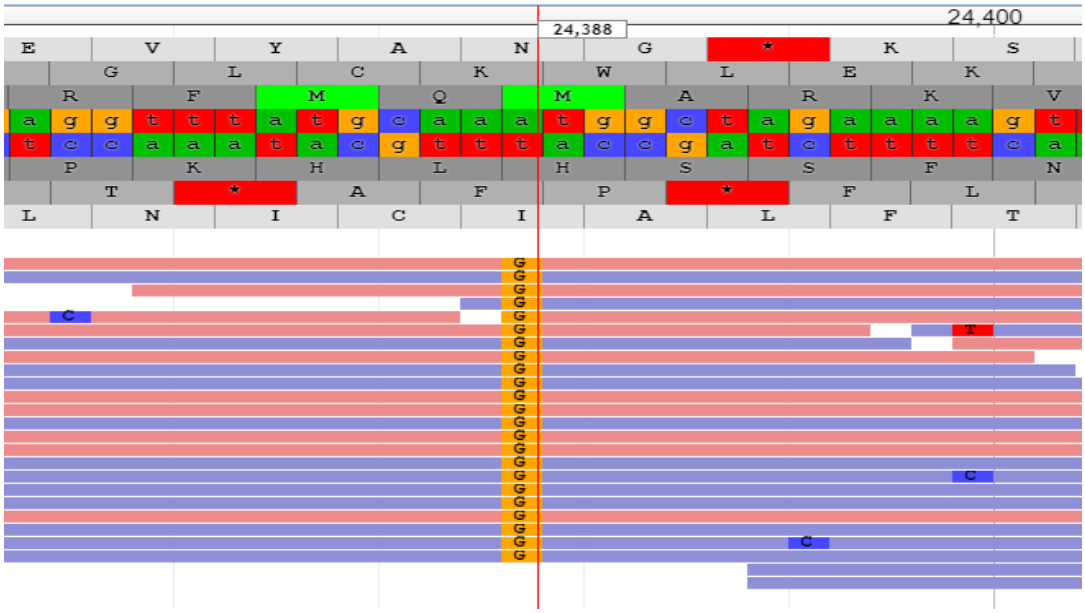


Figure 1: Reference genome vs Mutant-bacteria genome at position 24388.

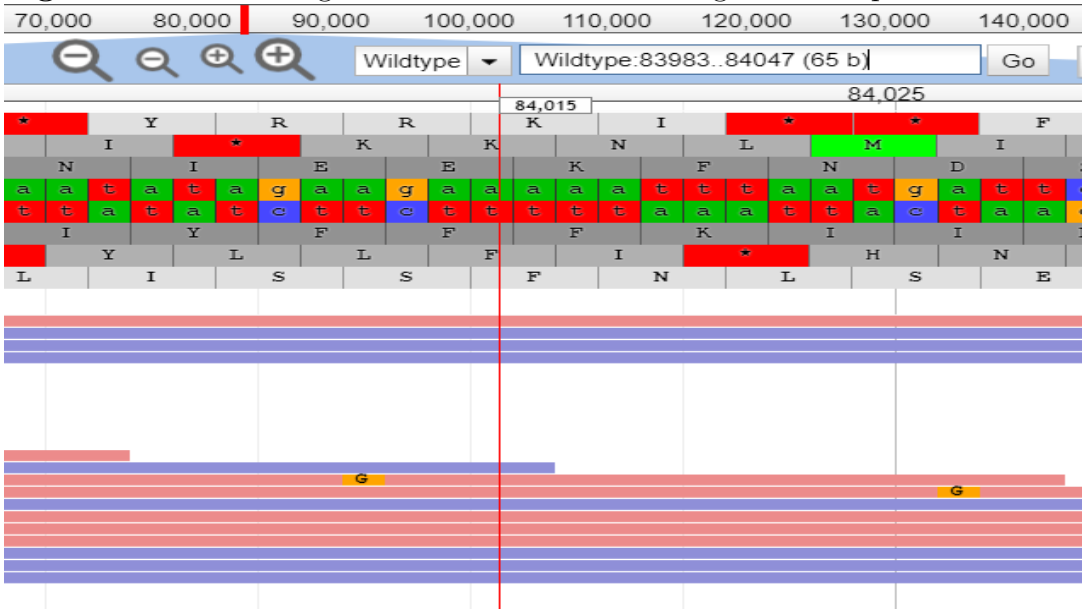


Figure 2: Reference genome vs Mutant-bacteria genome at position 84015.

5. (5 points) [Presentation gears up]: We are soon approaching the midpoint of this course and it is time to put your research hats on!

   (a) (1 point) Form a 2-member team (i.e., you and your team partner) for your upcoming paper presentation and course project. Provide your team name and team members (name/rollno.) here.
   (Bonus: Interdisciplinary teams will gain bonus points!)

   > **Solution:**
   > Team Name: BioBois
   > Member 1: Anoubhav Agarwaal ; BE16B002
   > Member 2: Rahul Nikam ; BT18D011

   (b) (2 points) Which research area and topic (within that area) would your team like to work on? You are welcome to choose the topic of most interest to you in bioinformatics, genomics or systems biology. A good list of topics is available at the Bioinformatics journal website; you could also check out other top-tier journals mentioned in Assignment 1 such as Cell Systems or conferences such as ISMB, RECOMB, ROCKY, or RSGDREAM.

   > **Solution:** Our team would like to work on the topic of **Gene Expression**. Particularly the statistical analysis of differential gene expression using Deep Learning.

   (c) (2 points) Give the reference of a research paper your team is planning to present. It's recommended (though not mandatory) to choose a paper in the same research area as your course project.

   > **Solution:** We are planning to present the paper: **Extracting a Biologically Relevant Latent Space from Cancer Transcriptomes with Variational Autoencoders**.[11]

References

[1] M. Mahajan, P. Nimbhorkar, and K. Varadarajan, "The planar k-means problem is np-hard," 2009.

[2] J. Garcia-Diaz, R. Menchaca-Mendez, R. Menchaca-Mendez, S. Pomares Hernández, J. C. Pérez-Sansalvador, and N. Lakouari, "Approximation algorithms for the vertex k-center problem: Survey and experimental evaluation," *IEEE Access*, vol. 7, pp. 109228–109245, 2019.

[3] Wang, Haizhou, and Mingzhou Song. "Ckmeans.1d.dp: Optimal k-means Clustering in One Dimension by Dynamic Programming." The R journal vol. 3,2 (2011): 29-33.

[4] D. Z. Chen, J. Li, and H. Wang, "Efficient algorithms for the one-dimensional k-center problem," *Theoretical Computer Science*, vol. 592, pp. 135 – 142, 2015.

[5] S. Dasgupta and P. M. Long, "Performance guarantees for hierarchical clustering," *Journal of Computer and System Sciences*, vol. 70, no. 4, pp. 555 – 569, 2005. Special Issue on COLT 2002.

[6] https://www.genomicseducation.hee.nhs.uk/blog/reference-genome-defining-human difference/

[7] https://en.wikipedia.org/wiki/Reference_genome/

[8] https://bitesizebio.com/38335/get-to-know-your-reference-genome-grch37-vs grch38/

[9] http://resources.qiagenbioinformatics.com/manuals/clcgenomicsworkbench/852/index.php?manu

[10] https://galaxyproject.github.io/training-material/topics/variant-analysis/tutorials/exome-seq/tutorial.html#mapped-reads postprocessing

[11] G. P. Way and C. S. Greene, "Extracting a biologically relevant latent space from cancer transcriptomes with variational autoencoders," *bioRxiv*, 2017.