Comparison of the complete genome sequences of COVID-۱۹ and Zika virus using BioPython

Ahmad Hafezi[۱], Zahra khamar[۱*]

[۱]Department of Biology, Faculty of science, Ferdowsi University of Mashhad, Mashhad, Iran

*Corresponding author: zahrakhamr@um.ac.ir

Abstract

Background: Bioinformatics has always been one of the important tools in biological research. Increasing in the biological data, triggered some difficulties in the data analysis. To solve these problems BioPython developed, and created a better platform for researchers to conduct their study in a short way just by writing appropriate BioPython codes.

Methods: In this research, bioinformatics sites such as NCBI were first used so that the genome sequences of both Zika and Corona viruses were completely stored in the Festa file. The PyCharm program was then used to apply BioPython codes and facilitate the speed of data analysis. Finally, the sequences were analyzed with BioPython codes to determine the GC percentage, length and number of amino acids that each sequence encodes.

Results: The results showed that, although the zika virus has a shorter DNA sequence, the amino acids ratio encoded by the whole genome is almost the same as COVID-۱۹ and the CG content of zika is more than coronavirus. After analyzing the sequence of both viruses, it was found that the three amino acids Thr, Ser and Leu are the three amino acids that both viruses had the most codons of their production in their genomes. In contrast, in both sequences, the amount of Met encoded by the entire genome is in fewer amounts.

Conclusions: In this research, the COVID-۱۹ and Zika virus whole genomes are analyzed with BioPython and compared with each other. As a result, both viruses have historically been major threats to human health, so knowing enough and having the necessary information about their genomes will enable scientists to find the right drug or vaccine to deal with them in a shorter time. In this case, obtaining the desired information in the shortest possible time using programming can improve the speed of work and help to perform calculations in a shorter time. In the meantime, Python programming enables us to obtain valuable

information about the desired strain in the shortest possible time in the next pandemic.

Keywords: BioPython, Data Analyzing, Bioinformatics, Sequence analyzing, COVID-۱۹

۱. Background

It was in ۱۹۹۱ when the first version of python published by Guido van Rossum (۱). After that Brad Chapman and Jeff Chang started to utilize the python for biological research and they published the BioPython for the first time on ۲۰۰۰(۲). Python is a programming language that is used extensively in academics and commercials and due to its user-friendly and its performance in all types of operating systems features, it has become the most well-known language programming in the world(۳). BioPython is an open-source compilation of Python tools for natural computation, made by a universal group of developers(۲,٤). BioPython has been significantly expanded during this century because using the appropriate modules and classes in biological analysis, enables us to solve complex bioinformatics problems. BioPython provides diverse types of tools to parse the bioinformatics files and accessibility to online websites such as NCBI(٥).

۱.۱. *COVID-۱۹*

Severe acute respiratory syndrome coronavirus (SARS-CoV-۲) emerged in ۲۰۱۹ in China. The coronaviridae family consist of four genera: alphacoronavirus, betacoronavirus, gammacoronavirus and deltacoronavirus that can infect mammals and birds. It is known that alpha- and betacoronaviruses usually infect humans, whereas gamma- and deltacoronaviruses

infect birds. Coronavirus is a positive sense RNA that the length of RNA is approximately ۳۰kb. The three types of proteins found in coronavirus include spike (s), membrane (M) and envelope (E) (Figure ۱). The S protein contains two domains, s۱ and s۲, which are essential for entry into host cells. A cell surface peptidase called ACE۲ hydrolyzes angiotensin II and is naturally high-expressed in the epithelium of the lung and small intestine recognized as a receptor for coronavirus. after binding ACE۲ and s protein, s protein is subsequently cleaved and activated by another cell surface protease MPRSS۲ and it leads to membrane fusion (٦).

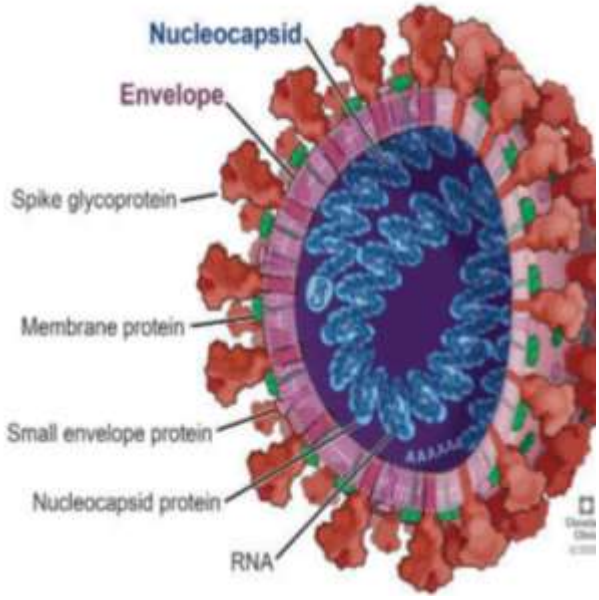


Figure ١: structure of coronavirus (٧)

١.١ *Zika virus*

In ١٩٤٧, the Zika virus was first isolated in a rhesus monkey (٨). Seven years later, the first human infection emerged in Nigeria (٩). Zika virus was isolated from some mosquitoes in Africa when researchers studied yellow fever (٨,١٠). The first official report of zika fever occurred in ٢٠٠٧ (١١) and a considerable epidemic emerged in French Polynesia in the South Pacific in ٢٠١٣ and ٢٠١٤ (١٢). It was evaluated that approximately ٣٠,٠٠٠ symptomatic infections were recorded (١٣,١٤). This epidemic outbreak in ٢٠١٤ in New Caledonia and initially disseminated in the Americas (Brazil in March)(١٥).

٢. Material and Methods

To have a better platform for running the BioPython codes, the PyCharm was downloaded on the https://www.jetbrains.com/pycharm/ site website. Afterwards, the whole genome of severe acute respiratory syndrome coronavirus ٢ and Zika virus were download from https://www.ncbi.nlm.nih.gov/ website with ID = NC_٠٤٥٥١٢.٢ and KX٨٩٣٨٥٥.١ respectively. These FASTA-sequence files were transferred to the PyCharm directory and used the below codes to analyze these sequences (١٦):

```
>>>from Bio.SeqRecord import SeqRecord

>>>from Bio.SeqIO import parse

>>>from Bio.Seq import Seq
>>>COVID_seq =
open("FASTA_sequence.fasta")
>>>detail = parse(COVID_seq, "fasta")
>>>for i in detail:
>>>     print("ID: %s " % i.id)
>>>     print("Name: %s " % i.name)
```

```
>>>     print("Description: %s " % i.description)
>>>     print("Sequence Data: %s " % i.seq)
```

This code separates the ID, Name, Description and the main sequence in the FASTA file. By falling into different parts of the FASTA component, it becomes easy to parse and analyze the main sequence. To find out the GC percentage in this sequence it is necessary to import the GC class from Bio.SeqUtils module.

```
>>>from Bio.SeqUtils import GC
>>> main_seq = []
>>>for i in detail:
>>>     main_seq = i.seq
>>>print(GC(main_seq))
```

To transcription this sequence, the following code was used. After that to translate this sequence the translate() should be applied. Also, the length of both RNA and amino acids sequences will be appeared by applying the len() command:

```
>>>from Bio.Seq import Seq
>>>from Bio.Seq import transcribe
>>>COVID_seq = open("FASTA_sequence.fasta")
>>>detail = parse(COVID_seq, "fasta")
>>>RNA = []
>>> main_seq = []
>>>for i in detail:
>>>     main_seq = i.seq
>>>     RNA = transcribe(main_seq)
>>>print(RNA)
>>>print(len(RNA))
>>>print(RNA.translate())
>>>print(len(RNA.translate()))
```

To indicate the standard codons table that uses for translation the CodonTable must be imported from Bio.Data:

```
>>>from Bio.Data import CodonTable
>>>codon_table = CodonTable.ambiguous_dna_by_name['Standard']
>>>print(codon_table)
```

Next, the number of each amino acids were calculated by writing below codes:

```
>>>from Bio.Data import IUPACData
>>>amino_acids =
IKGLYLPR*QTNQLSISCRSVL*TNFKIC
VAVTRLHA*CTHAV*LITNYCR*QDTS
NSSIFCRLL...MCKINFSSAIPM*F**LLR
RMTKKKKKKKKK
>>>type_amino = IUPACData.protein_letters
>>>count_amino = {}.fromkeys(type_amino, ۰)
>>>stop_codon = {}.fromkeys('stop_codon', ۰)
>>>for char in amino_acids:
>>>     if char in type_amino:
>>>          count_amino[char] += ۱
```

>>>for i in amino_acids:

>>>stop_codon=amino_acids.count('*')

>>>print(count_amino)

>>>print(IUPACData.protein_letters_۳to۱)

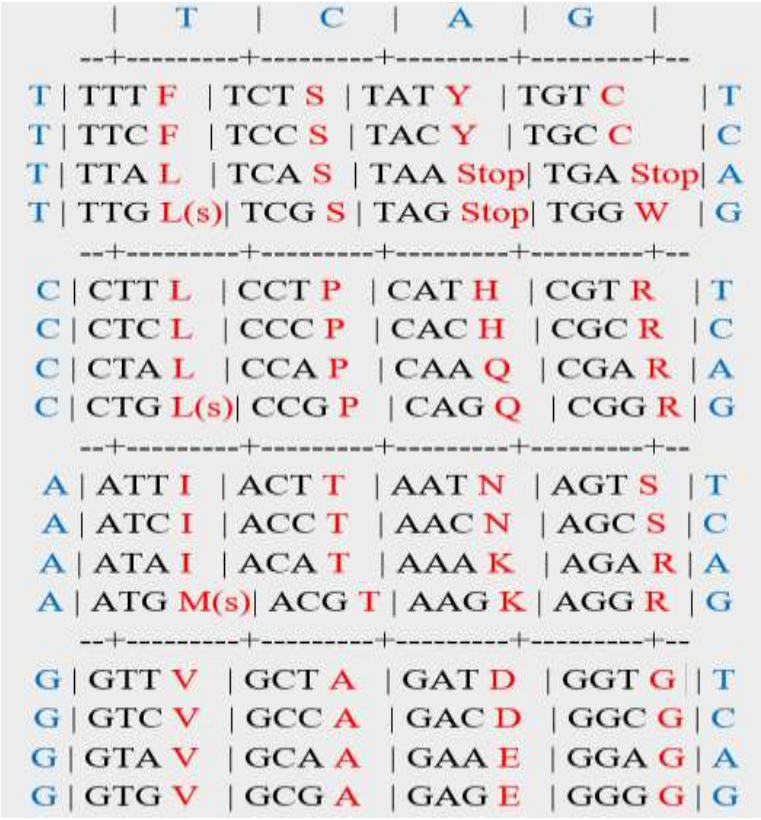
>>>print('stop_codon:',stop_codon)

All these steps were done for Zika virus genome. These codes were run in the PyCharm Console and then the results were checked.

۳. Results

In the first, the Codon Table was shown to understand what three types of basses in mRNA create a specific type of amino acids. In this table, four types of basses (T, C, A and G) are listed and the type of amino acid that will be created is shown beside of each the three basses with a special letter (Figure ۲). It is helpful to have an abbreviation of the name of twenty types of amino acids. Therefore, by using the IUPACData.protein_letters_۳to۱, all of the amino acids were shown with their abbreviation letters:

{'Ala': 'A', 'Cys': 'C', 'Asp': 'D', 'Glu': 'E', 'Phe': 'F', 'Gly': 'G', 'His': 'H', 'Ile': 'I', 'Lys': 'K', 'Leu': 'L', 'Met': 'M', 'Asn': 'N', 'Pro': 'P', 'Gln': 'Q', 'Arg': 'R', 'Ser': 'S', 'Thr': 'T', 'Val': 'V', 'Trp': 'W', 'Tyr': 'Y'}.



	T	C	A	G	
T	TTT F	TCT S	TAT Y	TGT C	T
T	TTC F	TCC S	TAC Y	TGC C	C
T	TTA L	TCA S	TAA Stop	TGA Stop	A
T	TTG L(s)	TCG S	TAG Stop	TGG W	G
C	CTT L	CCT P	CAT H	CGT R	T
C	CTC L	CCC P	CAC H	CGC R	C
C	CTA L	CCA P	CAA Q	CGA R	A
C	CTG L(s)	CCG P	CAG Q	CGG R	G
A	ATT I	ACT T	AAT N	AGT S	T
A	ATC I	ACC T	AAC N	AGC S	C
A	ATA I	ACA T	AAA K	AGA R	A
A	ATG M(s)	ACG T	AAG K	AGG R	G
G	GTT V	GCT A	GAT D	GGT G	T
G	GTC V	GCC A	GAC D	GGC G	C
G	GTA V	GCA A	GAA E	GGA G	A
G	GTG V	GCG A	GAG E	GGG G	G

Figure ٢: The standard genetic codons demonstrated in the PyCharm console (١٦)

٣.١. Covid ١٩ sequence analysis

In the first written code, all information which is available in the FASTA file, will be shown separately in the PyCharm console. ID and name indicate the specific identification which exclusively belongs to the COVID-١٩ sequence. The main information indicated in the description part gives complete data about the sequences. Finally, the sequence is imported into a separate part.

ID: NC_٠٤٥٥١٢.٢

Name: NC_٠٤٥٥١٢.٢

Description: NC_٠٤٥٥١٢.٢ severe acute respiratory syndrome coronavirus ٢ isolate Wuhan-Hu-١, complete genome

Sequence Data:
ATTAAAGGTTTATACCTTCCCAGGTA
ACAAACCAACCAACTTTCGATCTCTT
GTAGATCTGTTCTCTAAACGAACTTT
AAAATCTGTGTGGCTGTCACTCGGCT
GCATGCTTAGTGCATTTCCTATTCCTT
AGGGAACGTGGTTGACCTACACAGGT

GCCATCAAATTGGATG……...ACAAAG
ATCCAAATTTCTTAGGAGAATGACAA
AAAAAAAAAAAAAAAAAAAAAAA
AAAAAAA

GC content was evaluated by using BioPython according to the main sequence. For COVID-١٩, GC% = ٣٧.٩٧٢٧٧٨٦٥٠٩٧١٤٨ was estimated. BioPython used the transcribe module to replace the U nucleotide with the T nucleotide in the main sequence. And the whole sequence was completely translated. Also, the len() was used to estimates the length of the COVID-١٩ genome.

AUUAAAGGUUUAUACCUUCCCAGGU
AACAAACCAACCAACUUU.....CUUC
UUAGGAGAAUGACAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAA

Length = ٢٩٩٠٣ (bp)

After transcribing, all of the amino acids that are encoded by the whole genome were determined and its length was measured as well. In addition, the number of each amino acid was revealed beside the number of stop codons (*) in the PyCharm console (Figure ٣):

IKGLYLPR*QTNQLSISCRSVL*TNFKIC
VAVTRLHA*CTHAV*LITNYCR*QDTS
NSSIFCRLL...MCKINFSSAIPM*F**LLR
RMTKKKKKKKKK

Length= ٩٩٦٧ amino acids

stop_codon: ٧٧٤

{'A': ٣٧٥, 'C': ٦٣٥, 'D': ٢٩٠, 'E': ٢٧٠, 'F': ٥٩٣,

'G': ٣٩٤, 'H': ٣٣٢, 'I': ٤٣٦, 'K': ٤١٣, 'L': ٨٨٦,

'M': ١١٧, 'N': ٤٧٢, 'P': ٢٩٢, 'Q': ٣٢٥, 'R': ٥٥٨,

'S': ٨١٠, 'T': ٦٧٩, 'V': ٥٤٨, 'W': ٢٦٣, 'Y': ٥٠٥}.

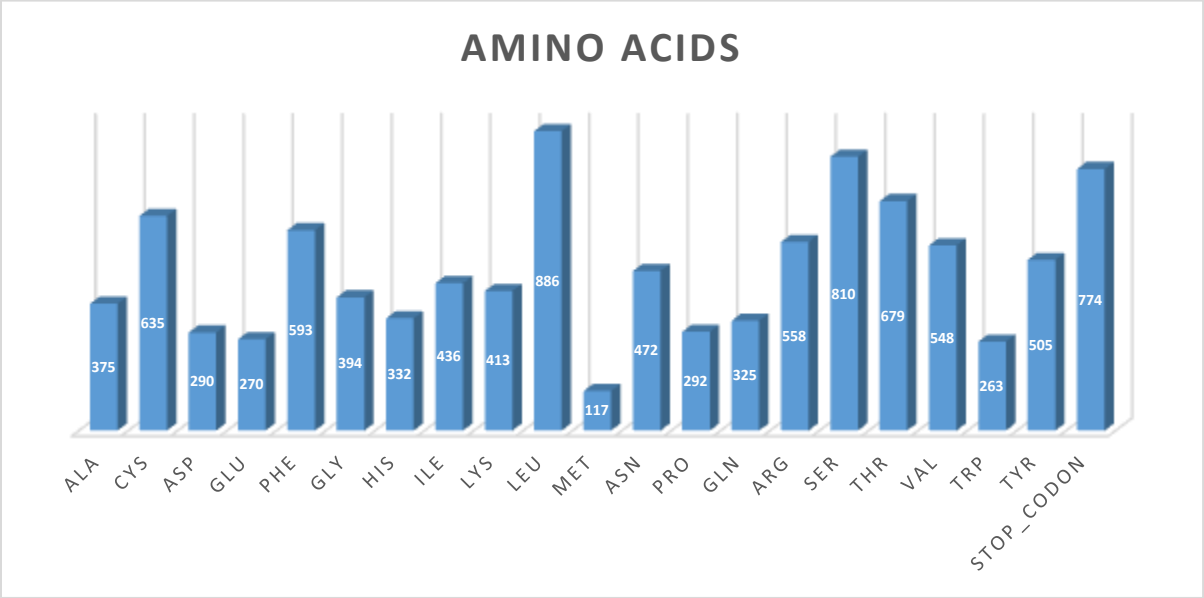


Figure ۳: The number of each amino acid and stop codons transcribed from the COVID-۱۹ whole genome

۳.۲. Zika virus sequence analysis

ID, Name, description and Sequence data in the FASTA file of the Zika virus fell into different sections and the main sequence was analyzed more by BioPython.

ID: KX۸۹۳۸۵۵.۱

Name: KX۸۹۳۸۵۵.۱

Description: KX۸۹۳۸۵۵.۱ Zika virus strain Zika virus/Homosapiens/VEN/UF-۲/۲۰۱۶, complete genome

Sequence Data:
AGTTGTTACTGTTGCTGACTCAGACT
GCGACAGTTCGAGTTT...GGGGAAATC
CATGGGTCTT

Transcribe:

AGUUGUUACUGUUGCUGACUCAGAC
UGCGACAGUUCGAGUUU...GGGGAAA
UCCAUGGGUCUU

Length = ۱۰۸۰۸ (bp)

%GC = ۵۱.۲۱۲۰۶۵۱۳۶۹۳۵۶۱

Translate:

SCYCC*LRLRQFEFEAKASNSINRFYFG
FGNESFWS*KTQKRNPEDSGLSIC...QK
RD*WLEETPRKTQNSILTLGKTRDSMS
FHHAGRQAQIAE*RRPVWGNPWV.

Length = ۳۶۰۲

stop_codon: ۲۱۳

{‘A’: ۱۶۷, ‘C’: ۱۳۷, ‘D’: ۷۳, ‘E’: ۱۶۸, ‘F’: ۷۰, ‘G’: ۲۷۰, ‘H’: ۹٤, ‘I’: ۸۷, ‘K’: ۱۹٤, ‘L’: ۳۳۷, ‘M’: ۸۹, ‘N’: ٥٥, ‘P’: ۲٤۳, ‘Q’: ۱۹٦, ‘R’: ۲۸٤, ‘S’: ۳۲٥, ‘T’: ۲٦۰, ‘V’: ۱۲۳, ‘W’: ۱۷۸, ‘Y’: ۳۷} (Figure ٤).

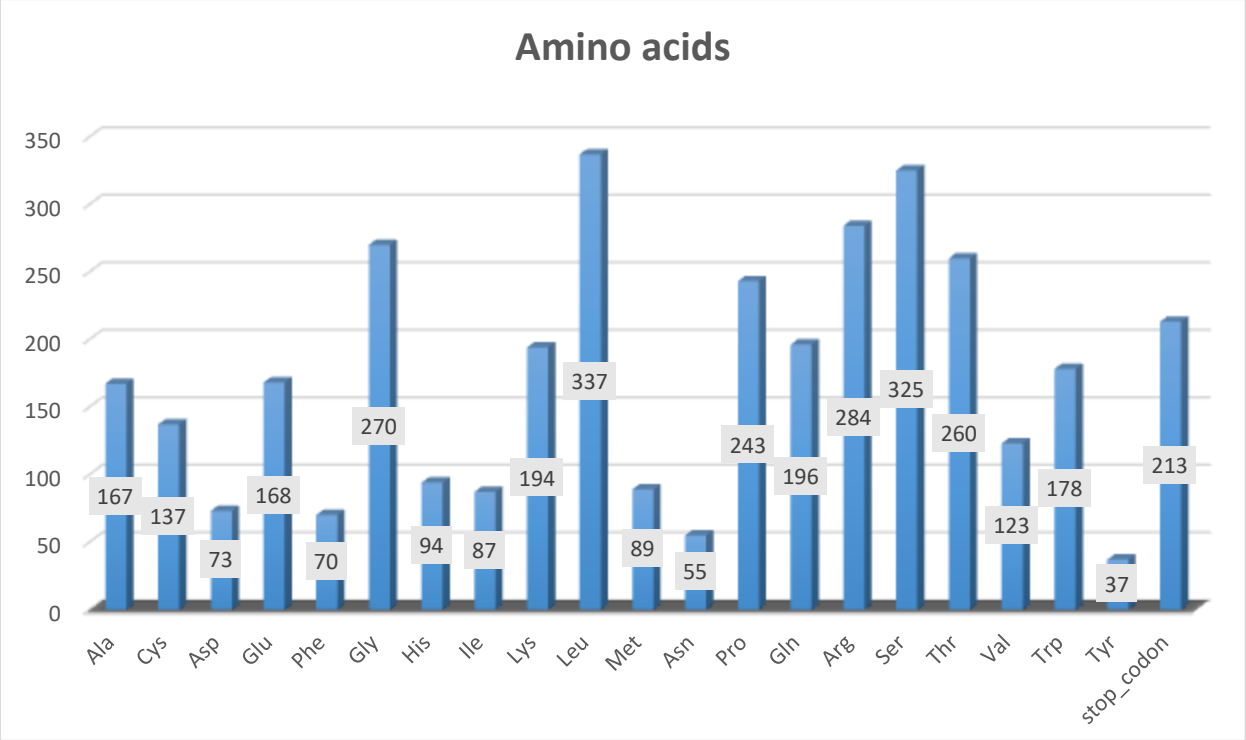


Figure ٤: The number of each amino acid transcribed from the Zika virus whole genome

By analyzing both genomes, the result showed that Thr, Ser and Leu are three of the most common amino acids in both sequences. Although Tyr and Asn have fewer codons in the Zika whole genome, they are two of the more typical amino acids in the COVID-۱۹ genome. Moreover, in both sequences, the amount of Met that is encoded by the whole genome is in fewer amounts. Another common feature in these results is related to the stop codons. As it is indicated, the number of stop codons in both genomes is in a high amount. Even though the length of the Zika genome is approximately three times shorter than that of the COVID-۱۹ genome, the ratio of amino acids encoded by the complete Zika genome is almost identical to the ratio of amino acids

encoded by the COVID-۱۹ complete genome.

predict the strains that can trigger the infection in advance. In this way, BioPython has become an open-source programming application that has tremendous potential for bioinformatics researchers.

٤. Conclusions

COVID-۱۹ and Zika are two viruses which triggered a large pandemic in the world. Therefore, it is necessary to analyze their genomes to extract the information that can help us to find an appropriate view to be prepared for the next pandemic. Today, BioPython has become one of the state-of-the-art language programming which can parse and analyze bioinformatics files in a short time. In this regard, Kukreja and Kumari conducted research to analyze the data related to brain cancer with BioPython. In the first step, they used the KEGG website to find the interaction between Glioma molecules. After the selection of related molecules, the GC content and the length of these genes were assessed (٥).

Overall, by parsing the sequence, the disease prediction will be done easily. For example, it is possible to analyze what happens if a nucleotide changes in a specific part of the genome with this awareness, some diseases can be treated early in diagnosis. Furthermore, by analyzing the infection strains genome it would be possible to

References

١. Van Rossum G. A brief timeline of Python. Hist Python. ٢٠٠٩;

٢. Chapman B, Chang J. Biopython: Python tools for computational biology. ACM Sigbio Newsl.

٢٠٠٠;٢٠(٢):١٥–٩.

٣. Oliphant TE. Python for scientific computing. Comput Sci Eng. ٢٠٠٧;٩(٣):١٠–٢٠.

٤. Cock PJA, Antao T, Chang JT, Chapman BA, Cox CJ, Dalke A, et al. Biopython: freely available Python tools for computational molecular biology and bioinformatics. Bioinformatics. ٢٠٠٩;٢٥(١١):١٤٢٢.

٥. Kukreja V, Kumari U. Data Analysis of Brain Cancer with Biopython.

٦. Hartenian E, Nandakumar D, Lari A, Ly M, Tucker JM, Glaunsinger BA. The molecular virology of coronaviruses. J Biol Chem. ٢٠٢٠;٢٩٥(٣٧):١٢٩١٠–٣٤.

٧. Bergmann CC, Silverman RH. COVID-١٩: Coronavirus replication, pathogenesis, and therapeutic strategies. Cleve Clin J Med.

۲۰۲۰;۸۷(٦):۳۲۱–۷.

۸. Dick GWA, Kitchen SF, Haddow AJ. Zika virus (I). Isolations and serological specificity. Trans R Soc Trop Med Hyg. ۱۹۵۲;٤٦(٥):٥۰۹–۲۰.

۹. Macnamara FN. Zika virus: a report on three cases of human infection during an epidemic of jaundice in Nigeria.

Trans R Soc Trop Med Hyg. ۱۹٥٤;٤۸(۲):۱۳۹–٤٥.

۱۰. Marchette NJ, Garcia R, Rudnick A. Isolation of Zika virus from Aedes aegypti mosquitoes in Malaysia. Am J Trop Med Hyg. ۱۹٦۹;۱۸(۳).

۱۱. Duffy MR, Chen TH, Hancock WT, Powers AM, Kool JL, Lanciotti RS, et al. Zika virus outbreak on Yap Island, federated states of Micronesia. N Engl J Med. ۲۰۰۹;۳٦۰(۲٤):۲٥۳٦–٤۳.

۱۲. Cao-Lormeau VM, Roche C, Teissier A, Robin E, Berry AL, Mallet HP, et al. Zika virus, French polynesia, South pacific, ۲۰۱۳. Emerg Infect Dis. ۲۰۱٤;۲۰(٦):۱۰۸٥.

۱۳. Musso D, Nilles EJ, Cao-Lormeau VM. Rapid spread of emerging Zika virus in the Pacific area. Clin Microbiol Infect. ۲۰۱٤;۲۰(۱۰):O٥۹٥–٦.

۱٤. Musso D, Gubler DJ. Zika virus. Clin Microbiol Rev. ۲۰۱٦;۲۹(۳):٤۸۷–٥۲٤.

۱٥. Campos GS, Bandeira AC, Sardi SI. Zika virus outbreak, bahia, brazil. Emerg Infect Dis. ۲۰۱٥;۲۱(۱۰):۱۸۸٥.

۱٦. Biopython - Quick Guide [Internet]. Available from: https://www.tutorialspoint.com/biopython/biopython_quick_guide.htm