REFERENCES

Abdullah, T., Faiza, M., Pant, P., Rayyan Akhtar, M., & Pant, P. (2016). An Analysis of Single Nucleotide Substitution in Genetic Codons - Probabilities and Outcomes. *Bioinformation*, *12*(3), 98–104. https://doi.org/10.6026/97320630012098

Adnan Shereen, M., Khan, S., Kazmi, A., Bashir, N., & Siddique, R. (2020). *COVID-19 infection: origin, transmission, and characteristics of human coronaviruses. Journal of Advanced Research.*doi:10.1016/j.jare.2020.03.005

Aftab, S. O., Ghouri, M. Z., Masood, M. U., Haider, Z., Khan, Z., Ahmad, A., & Munawar, N. (2020). *Analysis of SARS-CoV-2 RNA-dependent RNA polymerase as a potential therapeutic drug target using a computational approach. Journal of Translational Medicine, 18(1).* doi:10.1186/s12967-020-02439-0

Afzal, A. (2020). Molecular diagnostic technologies for COVID-19: Limitations and challenges. *Journal of Advanced Research*, *26*, 149–159. https://doi.org/10.1016/j.jare.2020.08.002

Astuti, I., & Ysrafil (2020). Severe Acute Respiratory Syndrome Coronavirus 2 (SARS-CoV-2): An overview of viral structure and host response. *Diabetes & metabolic syndrome*, *14*(4), 407–412. https://doi.org/10.1016/j.dsx.2020.04.020

Barba, M., Czosnek, H., & Hadidi, A. (2014). Historical perspective, development and applications of next-generation sequencing in plant virology. *Viruses*, *6*(1), 106–136. https://doi.org/10.3390/v6010106

Beek, M., Clements, D., Blankenberg, D., & Nekrutenko, A. (2021). Galaxy training: From NCBI's Sequence Read Archive (SRA) to galaxy: SARS-CoV-2 variant analysis (Galaxy Training Materials) Retrieved March 01, 2021, from https://training.galaxyproject.org/training-material/topics/variant-analysis/tutorials/sars-cov-2/tutorial.html

Behjati, S., & Tarpey, P. S. (2013). What is next generation sequencing?. *Archives of disease in childhood. Education and practice edition*, *98*(6), 236–238. https://doi.org/10.1136/archdischild-2013-304340

Bogner, P., Capua, I., Lipman, D. J., Cox, N. J., & others. (2006). *A global initiative on sharing avian flu data. Nature, 442(7106), 981–981.* doi:10.1038/442981a

Bohannan, Z. S., & Mitrofanova, A. (2019). Calling Variants in the Clinic: Informed Variant Calling Decisions Based on Biological, Clinical, and Laboratory Variables. *Computational and structural biotechnology journal*, *17*, 561–569. https://doi.org/10.1016/j.csbj.2019.04.002

Bolger, A. M., Lohse, M., & Usadel, B. (2014). Trimmomatic: a flexible trimmer for Illumina sequence data. *Bioinformatics (Oxford, England)*, *30*(15), 2114–2120. https://doi.org/10.1093/bioinformatics/btu170

Cascella, M., Rajnik, M., Cuomo, A., Dulebohn, S. C., & Di Napoli, R. (2021). Features, Evaluation, and Treatment of Coronavirus (COVID-19). In *StatPearls*. StatPearls Publishing.

Chaisson, M. J., Wilson, R. K., & Eichler, E. E. (2015). Genetic variation and the de novo assembly of human genomes. *Nature reviews. Genetics*, *16*(11), 627–640. https://doi.org/10.1038/nrg3933

Charre, C., Ginevra, C., Sabatier, M., Regue, H., Destras, G., Brun, S., Burfin, G., Scholtes, C., Morfin, F., Valette, M., Lina, B., Bal, A., & Josset, L. (2020). Evaluation of NGS-based approaches for SARS-CoV-2 whole genome characterisation. *Virus Evolution*, *6*(2). https://doi.org/10.1093/ve/veaa075

Chu, D., Hui, K., Gu, H., Ko, R., Krishnan, P., Ng, D....Poon, L. (2021). Introduction of ORF3a-Q57H SARS-CoV-2 Variant Causing Fourth Epidemic Wave of COVID-19, Hong Kong, China. *Emerging Infectious Diseases*, *27*(5), 1492-1495. https://doi.org/10.3201/eid2705.210015.

Cingolani, P., Platts, A., Wang, l., Coon, M., Nguyen, T., Wang, L., Land, S. J., Lu, X., & Ruden, D. M. (2012). A program for annotating and predicting the effects of single nucleotide polymorphisms, SnpEff: SNPs in the genome of Drosophila melanogaster strain w1118; iso-2; iso-3. *Fly*, *6*(2), 80–92. https://doi.org/10.4161/fly.19695

Cui, J., Li, F., & Shi, Z. L. (2019). Origin and evolution of pathogenic coronaviruses. *Nature reviews. Microbiology*, *17*(3), 181–192. https://doi.org/10.1038/s41579-018-0118-9

Cock, P. J., Fields, C. J., Goto, N., Heuer, M. L., & Rice, P. M. (2010). The Sanger FASTQ file format for sequences with quality scores, and the Solexa/Illumina FASTQ variants. *Nucleic acids research*, *38*(6), 1767–1771. https://doi.org/10.1093/nar/gkp1137

Cock, P. J., Antao, T., Chang, J. T., Chapman, B. A., Cox, C. J., Dalke, A., Friedberg, I., Hamelryck, T., Kauff, F., Wilczynski, B., & de Hoon, M. J. (2009). Biopython: freely available Python tools for computational molecular biology and bioinformatics. *Bioinformatics (Oxford, England)*, *25*(11), 1422–1423. https://doi.org/10.1093/bioinformatics/btp163

COVID-19 Genomics UK (COG-UK) consortiumcontact@cogconsortium.uk (2020). An integrated national scale SARS-CoV-2 genomic surveillance network. *The Lancet. Microbe*, *1*(3), e99–e100. https://doi.org/10.1016/S2666-5247(20)30054-9

COVID Symptom Study - Help slow the spread of COVID-19. (2021). https://covid.joinzoe.com/.

Danecek, P., & McCarthy, S. A. (2017). BCFtools/csq: haplotype-aware variant consequences. *Bioinformatics (Oxford, England)*, *33*(13), 2037–2039. https://doi.org/10.1093/bioinformatics/btx100

Decaro, N., & Lorusso, A. (2020). *Novel human coronavirus (SARS-CoV-2): A lesson from animal coronaviruses. Veterinary Microbiology, 244, 108693.* doi:10.1016/j.vetmic.2020.108693

Deng, X., Gu, W., Federman, S., du Plessis, L., Pybus, O. G., Faria, N. R., Wang, C., Yu, G., Bushnell, B., Pan, C. Y., Guevara, H., Sotomayor-Gonzalez, A., Zorn, K., Gopez, A., Servellita, V., Hsu, E., Miller, S., Bedford, T., Greninger, A. L., Roychoudhury, P., … Chiu, C. Y. (2020). Genomic surveillance reveals multiple introductions of SARS-CoV-2 into Northern California. *Science (New York, N.Y.)*, *369*(6503), 582–587. https://doi.org/10.1126/science.abb9263

Domingo E. (2020). Molecular basis of genetic variation of viruses: error-prone replication. *Virus as Populations*, 35–71. https://doi.org/10.1016/B978-0-12-816331-3.00002-7

Drew, D. A., Nguyen, L. H., Steves, C. J., Menni, C., Freydin, M., Varsavsky, T., Sudre, C. H., Cardoso, M. J., Ourselin, S., Wolf, J., Spector, T. D., Chan, A. T., & COPE Consortium (2020). Rapid implementation of mobile technology for real-time epidemiology of COVID-19. *Science (New York, N.Y.)*, *368*(6497), 1362–1367. https://doi.org/10.1126/science.abc0473

Elbe, S., & Buckland-Merrett, G. (2017). Data, disease and diplomacy: GISAID's innovative contribution to global health. *Global challenges (Hoboken, NJ)*, *1*(1), 33–46. https://doi.org/10.1002/gch2.1018

Gaudin, M., & Desnues, C. (2018). Hybrid Capture-Based Next Generation Sequencing and Its Application to Human Infectious Diseases. *Frontiers in microbiology*, *9*, 2924. https://doi.org/10.3389/fmicb.2018.02924

Genomic epidemiology data infrastructure needs for SARS-CoV-2: Modernizing pandemic response strategies. (2020). Washington, DC: The National Academies Press.

Gibson, P. G., Qin, L., & Puah, S. H. (2020). COVID-19 acute respiratory distress syndrome (ARDS): clinical features and differences from typical pre-COVID-19 ARDS. *The Medical journal of Australia*, *213*(2), 54–56.e1. https://doi.org/10.5694/mja2.50674

Gilchrist, C. A., Turner, S. D., Riley, M. F., Petri, W. A., & Hewlett, E. L. (2015). *Whole-Genome Sequencing in Outbreak Analysis. Clinical Microbiology Reviews, 28(3), 541–563.* doi:10.1128/cmr.00075-13

Gnirke, A., Melnikov, A., Maguire, J., Rogov, P., LeProust, E. M., Brockman, W., Fennell, T., Giannoukos, G., Fisher, S., Russ, C., Gabriel, S., Jaffe, D. B., Lander, E. S., & Nusbaum, C. (2009). Solution hybrid selection with ultra-long oligonucleotides for massively parallel targeted sequencing. *Nature biotechnology*, *27*(2), 182–189. https://doi.org/10.1038/nbt.1523

Gómez, C. E., Perdiguero, B., & Esteban, M. (2021). Emerging SARS-CoV-2 Variants and Impact in Global Vaccination Programs against SARS-CoV-2/COVID-19. *Vaccines*, *9*(3), 243. https://doi.org/10.3390/vaccines9030243

Gorbalenya, A. E., Baker, S. C., Baric, R. S., De Groot, R. J., Drosten, C., Gulyaeva, A. A., . . . Ziebuhr, J. (2020). Severe acute Respiratory syndrome-related CORONAVIRUS: The species and its Viruses – a statement of THE Coronavirus study group. doi:10.1101/2020.02.07.937862

Gunadi, Wibawa, H., Marcellus, Hakim, M. S., Daniwijaya, E. W., Rizki, L. P., . . . Wibawa, T. (2020). Full-length genome characterization and phylogenetic analysis OF SARS-COV-2 virus strains from Yogyakarta and central Java, Indonesia. *PeerJ, 8*. doi:10.7717/peerj.10575

Guo YR, Cao QD, Hong ZS, Tan YY, Chen SD, Jin HJ, Tan KS, Wang DY, Yan Y (2020) The origin, transmission and clinical therapies on coronavirus disease 2019 (COVID-19) outbreak— an update on the status. Mil Med Res 7(1):11. https://doi.org/10.1186/s40779-020-00240-0

Gong, Y.-N., Yang, S.-L., Chen, G.-W., Chen, Y.-W., Huang, Y.-C., Ning, H.-C., & Tsao, K.-C. (2017). *A metagenomics study for the identification of respiratory viruses in mixed clinical specimens: an application of the iterative mapping approach. Archives of Virology, 162(7), 2003–2012.* doi:10.1007/s00705-017-3367-4

Graham, M. S., Sudre, C. H., May, A., Antonelli, M., Murray, B., Varsavsky, T., Kläser, K., Canas, L. S., Molteni, E., Modat, M., Drew, D. A., Nguyen, L. H., Polidori, L., Selvachandran, S., Hu, C., Capdevila, J., COVID-19 Genomics UK (COG-UK) Consortium, Hammers, A., Chan, A. T., Wolf, J., … Ourselin, S. (2021). Changes in symptomatology, reinfection, and transmissibility associated with the SARS-CoV-2 variant B.1.1.7: an ecological study. *The Lancet. Public health*, *6*(5), e335–e345. https://doi.org/10.1016/S2468-2667(21)00055-4

Grüning, B., Chilton, J., Köster, J., Dale, R., Soranzo, N., van den Beek, M., Goecks, J., Backofen, R., Nekrutenko, A., & Taylor, J. (2018). Practical Computational Reproducibility in the Life Sciences. *Cell systems*, *6*(6), 631–635. https://doi.org/10.1016/j.cels.2018.03.014

Grüning, B., Dale, R., Sjödin, A., Chapman, B. A., Rowe, J., Tomkins-Tinch, C. H., Valieris, R., Köster, J., & Bioconda Team (2018). Bioconda: sustainable and comprehensive software distribution for the life sciences. *Nature methods*, *15*(7), 475–476. https://doi.org/10.1038/s41592-018-0046-7

He, B., Zhang, Y., Xu, L., Yang, W., Yang, F., Feng, Y., Xia, L., Zhou, J., Zhen, W., Feng, Y., Guo, H., Zhang, H., & Tu, C. (2014). Identification of diverse alphacoronaviruses and genomic characterization of a novel severe acute respiratory syndrome-like coronavirus from bats in China. *Journal of virology*, *88*(12), 7070–7082. https://doi.org/10.1128/JVI.00631-14

Head, S. R., Komori, H. K., LaMere, S. A., Whisenant, T., Van Nieuwerburgh, F., Salomon, D. R., & Ordoukhanian, P. (2014). Library construction for next-generation sequencing: overviews and challenges. *BioTechniques*, *56*(2), 61–passim. https://doi.org/10.2144/000114133

Houldcroft, C. J., Beale, M. A., & Breuer, J. (2017). Clinical and biological insights from viral genome sequencing. *Nature reviews. Microbiology*, *15*(3), 183–192. https://doi.org/10.1038/nrmicro.2016.182

Hourdel, V., Kwasiborski, A., Balière, C., Matheus, S., Batéjat, C. F., Manuguerra, J. C., Vanhomwegen, J., & Caro, V. (2020). Rapid Genomic Characterization of SARS-CoV-2 by Direct Amplicon-Based Sequencing Through Comparison of MinION and Illumina iSeq100TM System. *Frontiers in microbiology*, *11*, 571328. https://doi.org/10.3389/fmicb.2020.571328

Illumina (2020). Enrichment Workflow for Detecting Coronavirus Using Illumina NGS Systems. Accessed January 29, 2021.

Jiang, S., Shi, Z., Shu, Y., Song, J., Gao, G. F., Tan, W., & Guo, D. (2020). *A distinct name is needed for the new coronavirus. The Lancet.* doi:10.1016/s0140-6736(20)30419-0

Johnson A. D. (2010). An extended IUPAC nomenclature code for polymorphic nucleic acids. *Bioinformatics (Oxford, England)*, *26*(10), 1386–1389. https://doi.org/10.1093/bioinformatics/btq098

Kathiresan, N., Temanni, R., Almabrazi, H., Syed, N., Jithesh, P. V., & Al-Ali, R. (2017). *Accelerating next generation sequencing data analysis with system level optimizations. Scientific Reports, 7(1).* doi:10.1038/s41598-017-09089-1

Khailany, R. A., Safdar, M., & Ozaslan, M. (2020). Genomic characterization of a novel SARS-CoV-2.

Gene reports, *19*, 100682. https://doi.org/10.1016/j.genrep.2020.100682

Kim, D., Paggi, J. M., Park, C., Bennett, C., & Salzberg, S. L. (2019). Graph-based genome alignment and genotyping with HISAT2 and HISAT-genotype. Nature Biotechnology, 37(8), 907–915.doi:10.1038/s41587-019-0201-4

Koboldt, D. C., Steinberg, K. M., Larson, D. E., Wilson, R. K., & Mardis, E. R. (2013). The next-generation sequencing revolution and its impact on genomics. *Cell*, *155*(1), 27–38. https://doi.org/10.1016/j.cell.2013.09.006

Koyama, T., Platt, D., & Parida, L. (2020). Variant analysis of SARS-CoV-2 genomes. *Bulletin of the World Health Organization*, *98*(7), 495–504. https://doi.org/10.2471/BLT.20.253591

Krusche, P., Trigg, L., Boutros, P. C., Mason, C. E., De La Vega, F. M., Moore, B. L., Gonzalez-Porta, M., Eberle, M. A., Tezak, Z., Lababidi, S., Truty, R., Asimenos, G., Funke, B., Fleharty, M., Chapman, B. A., Salit, M., Zook, J. M., & Global Alliance for Genomics and Health Benchmarking Team (2019). Best practices for benchmarking germline small-variant calls in human genomes. *Nature biotechnology*, *37*(5), 555–560. https://doi.org/10.1038/s41587-019-0054-x

Kumar, R., Nagpal, S., Kaushik, S., & Mendiratta, S. (2020). COVID-19 diagnostic approaches: different roads to the same destination. *VirusDisease*, *31*(2), 97–105. https://doi.org/10.1007/s13337-020-00599-7

Kumar, S., Nyodu, R., Maurya, V. K., & Saxena, S. K. (2020). Morphology, Genome Organization, Replication, and Pathogenesis of Severe Acute Respiratory Syndrome Coronavirus 2 (SARS-CoV-2). *Coronavirus Disease 2019 (COVID-19): Epidemiology, Pathogenesis, Diagnosis, and Therapeutics* , 23–31. https://doi.org/10.1007/978-981-15-4814-7_3

Kunal, S., Aditi, Gupta, K., & Ish, P. (2021). COVID-19 variants in India : potential role in second wave and impact on vaccination. *Heart & Lung*, Advance online publication. https://doi.org/10.1016/j.hrtlng.2021.05.008

Kustin, T., Ling, G., Sharabi, S., Ram, D., Friedman, N., Zuckerman, N., … Mandelboim, M. (2019). *A method to identify respiratory virus infections in clinical samples using next-generation sequencing. Scientific Reports, 9(1).* doi:10.1038/s41598-018-37483-w

Lee, N., Cao, B., Ke, C., Lu, H., Hu, Y., Tam, C., Ma, R., Guan, D., Zhu, Z., Li, H., Lin, M., Wong, R., Yung, I., Hung, T. N., Kwok, K., Horby, P., Hui, D., Chan, M., & Chan, P. (2017). IFITM3, TLR3, and CD55 Gene SNPs and Cumulative Genetic Risks for Severe Outcomes in Chinese Patients With H7N9/H1N1pdm09 Influenza. *The Journal of infectious diseases*, *216*(1), 97–104. https://doi.org/10.1093/infdis/jix235

Li, H., Handsaker, B., Wysoker, A., Fennell, T., Ruan, J., Homer, N., Marth, G., Abecasis, G., Durbin, R., & 1000 Genome Project Data Processing Subgroup (2009). The Sequence Alignment/Map format and SAMtools. *Bioinformatics (Oxford, England)*, *25*(16), 2078–2079. https://doi.org/10.1093/bioinformatics/btp352

Li, H., & Durbin, R. (2009). Fast and accurate short read alignment with Burrows-Wheeler transform. *Bioinformatics (Oxford, England)*, *25*(14), 1754–1760. https://doi.org/10.1093/bioinformatics/btp324

Lo, S. W., & Jamrozy, D. (2020). Genomics and epidemiological surveillance. *Nature reviews. Microbiology*, *18*(9), 478. https://doi.org/10.1038/s41579-020-0421-0

Ludwig, S., & Zarbock, A. (2020). Coronaviruses and SARS-CoV-2: A Brief Overview. *Anesthesia and analgesia*, *131*(1), 93–96. https://doi.org/10.1213/ANE.0000000000004845

Mamanova, L., Coffey, A. J., Scott, C. E., Kozarewa, I., Turner, E. H., Kumar, A., … Turner, D. J. (2010). *Target-enrichment strategies for next-generation sequencing. Nature Methods, 7(2), 111–118.* doi:10.1038/nmeth.1419

Mangul, S., Martin, L. S., Hill, B. L., Lam, A. K., Distler, M. G., Zelikovsky, A., Eskin, E., & Flint, J. (2019). Systematic benchmarking of omics computational tools. *Nature communications*, *10*(1), 1393. https://doi.org/10.1038/s41467-019-09406-4

Mercatelli, D., & Giorgi, F. M. (2020). Geographic and Genomic Distribution of SARS-CoV-2

Mutations. *Frontiers in microbiology*, *11*, 1800. https://doi.org/10.3389/fmicb.2020.01800

McAuley, A. J., Kuiper, M. J., Durr, P. A., Bruce, M. P., Barr, J., Todd, S., Au, G. G., Blasdell, K., Tachedjian, M., Lowther, S., Marsh, G. A., Edwards, S., Poole, T., Layton, R., Riddell, S.-J., Drew, T. W., Druce, J. D., Smith, T. R., Broderick, K. E., & Vasan, S. S. (2020). Experimental and in silico evidence suggests vaccines are unlikely to be affected by D614G mutation in SARS-CoV-2 spike protein. *Npj Vaccines*, *5*(1). https://doi.org/10.1038/s41541-020-00246-8

Naqvi, A., Fatima, K., Mohammad, T., Fatima, U., Singh, I. K., Singh, A., Atif, S. M., Hariprasad, G., Hasan, G. M., & Hassan, M. I. (2020). Insights into SARS-CoV-2 genome, structure, evolution, pathogenesis and therapies: Structural genomics approach. *Biochimica et biophysica acta. Molecular basis of disease*, *1866*(10), 165878. https://doi.org/10.1016/j.bbadis.2020.165878

Naming convention. (n.d.). Retrieved March 02, 2021, from https://support.illumina.com/help/BaseSpace_OLH_009008/Content/Source/Informatics/BS/NamingConvention_FASTQ-files-swBS.htm

Ng, P. C., & Kirkness, E. F. (2010). Whole genome sequencing. *Methods in molecular biology (Clifton, N.J.)*, *628*, 215–226. https://doi.org/10.1007/978-1-60327-367-1_12

No, J. S., Kim, W.-K., Cho, S., Lee, S.-H., Kim, J.-A., Lee, D., … Song, J.-W. (2019). *Comparison of targeted next-generation sequencing for whole-genome sequencing of Hantaan orthohantavirus in Apodemus agrarius lung tissues. Scientific Reports, 9(1).* doi:10.1038/s41598-019-53043-2

Nogales, A., & L DeDiego, M. (2019). Host Single Nucleotide Polymorphisms Modulating Influenza A Virus Disease in Humans. *Pathogens (Basel, Switzerland)*, *8*(4), 168. https://doi.org/10.3390/pathogens8040168

Novianti, A. N., Rahardjo, K., Prasetya, R. R., Nastri, A. M., Dewantari, J. R., Rahardjo, A. P., Estoepangestie, A., Shimizu, Y. K., Poetranto, E. D., Soegiarto, G., Mori, Y., & Shimizu, K. (2019). Whole-Genome Sequence of an Avian Influenza A/H9N2 Virus Isolated from an Apparently Healthy Chicken at a Live-Poultry Market in Indonesia. *Microbiology resource announcements*, *8*(17), e01671-18. https://doi.org/10.1128/MRA.01671-18

Oude Munnink, B. B., Nieuwenhuijse, D. F., Stein, M., O'Toole, Á., Haverkate, M., … Koopmans, M. (2020). *Rapid SARS-CoV-2 whole-genome sequencing and analysis for informed public health decision-making in the Netherlands. Nature Medicine.* doi:10.1038/s41591-020-0997-y

Park WB, Kwon NJ, Choi SJ, Kang CK, Choe PG, Kim JY, Yun J, Lee GW, Seong MW, Kim NJ, Seo JS, Oh MD (2020) Virus isolation from the first patient with SARS-CoV-2 in Korea. J Korean Med Sci 35(7):e84. https://doi.org/10.3346/jkms.2020.35.e84

Parvin, R., Heenemann, K., Halami, M. Y., Chowdhury, E. H., Islam, M. R., & Vahlenkamp, T. W. (2014). *Full-genome analysis of avian influenza virus H9N2 from Bangladesh reveals internal gene reassortments with two distinct highly pathogenic avian influenza viruses. Archives of Virology, 159(7), 1651–1661.* doi:10.1007/s00705-014-1976-8

Picard. (2021). Retrieved March 02, 2021, from https://broadinstitute.github.io/picard/

Plante, J. A., Liu, Y., Liu, J., Xia, H., Johnson, B. A., Lokugamage, K. G., … Shi, P.-Y. (2020). *Spike mutation D614G alters SARS-CoV-2 fitness. Nature.* doi:10.1038/s41586-020-2895-3

Qin D. (2019). Next-generation sequencing and its clinical application. *Cancer biology & medicine*, *16*(1), 4–10. https://doi.org/10.20892/j.issn.2095-3941.2018.0055

Quinlan, A. R., & Hall, I. M. (2010). BEDTools: a flexible suite of utilities for comparing genomic features. *Bioinformatics (Oxford, England)*, *26*(6), 841–842. https://doi.org/10.1093/bioinformatics/btq033

Rambaut, A., Holmes, E. C., O'Toole, Á., Hill, V., McCrone, J. T., Ruis, C., … Pybus, O. G. (2020). *A dynamic nomenclature proposal for SARS-CoV-2 lineages to assist genomic epidemiology. Nature Microbiology.* doi:10.1038/s41564-020-0770-5

Saha, P., Banerjee, A. K., Tripathi, P. P., Srivastava, A. K., & Ray, U. (2020). A virus that has gone viral: amino acid mutation in S protein of Indian isolate of Coronavirus COVID-19 might impact receptor binding, and thus, infectivity. *Bioscience reports*, *40*(5), BSR20201312.

https://doi.org/10.1042/BSR20201312
Schbath, S., Martin, V., Zytnicki, M., Fayolle, J., Loux, V., & Gibrat, J. F. (2012). Mapping reads on a genomic sequence: an algorithmic overview and a practical comparative analysis. *Journal of computational biology : a journal of computational molecular cell biology*, *19*(6), 796–813. https://doi.org/10.1089/cmb.2012.0022
Shen, W., Le, S., Li, Y., & Hu, F. (2016). SeqKit: A Cross-Platform and Ultrafast Toolkit for FASTA/Q File Manipulation. *PloS one*, *11*(10), e0163962. https://doi.org/10.1371/journal.pone.0163962
Shu, Y., & McCauley, J. (2017). GISAID: Global initiative on sharing all influenza data - from vision to reality. *Euro surveillance : bulletin Europeen sur les maladies transmissibles = European communicable disease bulletin*, *22*(13), 30494. https://doi.org/10.2807/1560-7917.ES.2017.22.13.30494
Sims, D., Sudbery, I., Ilott, N. E., Heger, A., & Ponting, C. P. (2014). Sequencing depth and coverage: key considerations in genomic analyses. *Nature reviews. Genetics*, *15*(2), 121–132. https://doi.org/10.1038/nrg3642
Singer, J. B., Thomson, E. C., Hughes, J., Aranday-Cortes, E., McLauchlan, J., da Silva Filipe, A., Tong, L., Manso, C. F., Gifford, R. J., Robertson, D. L., Barnes, E., Ansari, M. A., Mbisa, J. L., Bibby, D. F., Bradshaw, D., & Smith, D. (2019). Interpreting Viral Deep Sequencing Data with GLUE. *Viruses*, *11*(4), 323. https://doi.org/10.3390/v11040323
Slatko, B. E., Gardner, A. F., & Ausubel, F. M. (2018). Overview of Next-Generation Sequencing Technologies. *Current protocols in molecular biology*, *122*(1), e59. https://doi.org/10.1002/cpmb.59
Spheres. (2020, July 27). Retrieved March 05, 2021, from https://www.cdc.gov/coronavirus/2019-ncov/covid-data/spheres.html
Sternke, M., Tripp, K. W., & Barrick, D. (2020). *The use of consensus sequence information to engineer stability and activity in proteins. Methods in Enzymology, 149–179.* doi:10.1016/bs.mie.2020.06.001
Trivedi, U. H., Cézard, T., Bridgett, S., Montazam, A., Nichols, J., Blaxter, M., & Gharbi, K. (2014). Quality control of next-generation sequencing data without a reference. *Frontiers in genetics*, *5*, 111. https://doi.org/10.3389/fgene.2014.00111
Udugama, B., Kadhiresan, P., Kozlowski, H. N., Malekjahani, A., Osborne, M., Li, V., Chen, H., Mubareka, S., Gubbay, J. B., & Chan, W. (2020). Diagnosing COVID-19: The Disease and Tools for Detection. *ACS nano*, *14*(4), 3822–3835. https://doi.org/10.1021/acsnano.0c02624
van Nimwegen, K. J., van Soest, R. A., Veltman, J. A., Nelen, M. R., van der Wilt, G. J., Vissers, L. E., & Grutters, J. P. (2016). Is the $1000 Genome as Near as We Think? A Cost Analysis of Next-Generation Sequencing. *Clinical chemistry*, *62*(11), 1458–1464. https://doi.org/10.1373/clinchem.2016.258632
Volz, E., Mishra, S., Chand, M., Barrett, J. C., Johnson, R., Geidelberg, L., . . . Ferguson, N. M. (2021). Transmission of sars-cov-2 Lineage B.1.1.7 in England: Insights from linking epidemiological and genetic data. doi:10.1101/2020.12.30.20249034
Wang, L., Wang, Y., Ye, D., & Liu, Q. (2020). Review of the 2019 novel coronavirus (SARS-CoV-2) based on current evidence. *International journal of antimicrobial agents*, *55*(6), 105948. https://doi.org/10.1016/j.ijantimicag.2020.105948
Wang, Z., Gerstein, M., & Snyder, M. (2009). RNA-Seq: a revolutionary tool for transcriptomics. *Nature reviews. Genetics*, *10*(1), 57–63. https://doi.org/10.1038/nrg2484
Weber, L. M., Saelens, W., Cannoodt, R., Soneson, C., Hapfelmeier, A., Gardner, P. P., Boulesteix, A. L., Saeys, Y., & Robinson, M. D. (2019). Essential guidelines for computational method benchmarking. *Genome biology*, *20*(1), 125. https://doi.org/10.1186/s13059-019-1738-8
WHO Coronavirus Disease (COVID-19) Dashboard. (n.d.). Retrieved February 23, 2021, from https://covid19.who.int/
Wilm, A., Aw, P. P., Bertrand, D., Yeo, G. H., Ong, S. H., Wong, C. H., Khor, C. C., Petric, R., Hibberd, M. L., & Nagarajan, N. (2012). LoFreq: a sequence-quality aware, ultra-sensitive variant caller

for uncovering cell-population heterogeneity from high-throughput sequencing datasets. *Nucleic acids research*, *40*(22), 11189–11201. https://doi.org/10.1093/nar/gks918

Woo, P. C., Lau, S. K., Lam, C. S., Lau, C. C., Tsang, A. K., Lau, J. H., Bai, R., Teng, J. L., Tsang, C. C., Wang, M., Zheng, B. J., Chan, K. H., & Yuen, K. Y. (2012). Discovery of seven novel Mammalian and avian coronaviruses in the genus deltacoronavirus supports bat coronaviruses as the gene source of alphacoronavirus and betacoronavirus and avian coronaviruses as the gene source of gammacoronavirus and deltacoronavirus. *Journal of virology*, *86*(7), 3995–4008. https://doi.org/10.1128/JVI.06540-11

World Health Organization. (2021). *Tracking SARS-CoV-2 variants*. World Health Organization. https://www.who.int/en/activities/tracking-SARS-CoV-2-variants/.

Wu, F., Zhao, S., Yu, B., Chen, Y. M., Wang, W., Song, Z. G., Hu, Y., Tao, Z. W., Tian, J. H., Pei, Y. Y., Yuan, M. L., Zhang, Y. L., Dai, F. H., Liu, Y., Wang, Q. M., Zheng, J. J., Xu, L., Holmes, E. C., & Zhang, Y. Z. (2020). A new coronavirus associated with human respiratory disease in China. *Nature*, *579*(7798), 265–269. https://doi.org/10.1038/s41586-020-2008-3

Wu, D., Wu, T., Liu, Q., & Yang, Z. (2020). The SARS-CoV-2 outbreak: What we know. *International journal of infectious diseases : IJID : official publication of the International Society for Infectious Diseases*, *94*, 44–48. https://doi.org/10.1016/j.ijid.2020.03.004

Yang, Y., Walls, S. D., Gross, S. M., Schroth, G. P., Jarman, R. G., & Hang, J. (2018). Targeted Sequencing of Respiratory Viruses in Clinical Specimens for Pathogen Identification and Genome-Wide Analysis. *Methods in molecular biology (Clifton, N.J.)*, *1838*, 125–140. https://doi.org/10.1007/978-1-4939-8682-8_10

Zhang L, Bai W, Yuan N, Du Z (2019). Comprehensively benchmarking applications for detecting copy number variation. PLOS Computational Biology 15(9): e1007367. https://doi.org/10.1371/journal.pcbi.1007367

Zhang, Z., & Gerstein, M. (2003). Patterns of nucleotide substitution, insertion and deletion in the human genome inferred from pseudogenes. *Nucleic acids research*, *31*(18), 5338–5348. https://doi.org/10.1093/nar/gkg745

Zhang, L., Jackson, C. B., Mou, H., Ojha, A., Rangarajan, E. S., Izard, T., Farzan, M., & Choe, H. (2020). The D614G mutation in the SARS-CoV-2 spike protein reduces S1 shedding and increases infectivity. bioRxiv : the preprint server for biology, 2020.06.12.148726. https://doi.org/10.1101/2020.06.12.148726

Zheng J. (2020). SARS-CoV-2: an Emerging Coronavirus that Causes a Global Threat. *International journal of biological sciences*, *16*(10), 1678–1685. https://doi.org/10.7150/ijbs.45053

APPENDICES

Automated Bash Script to run the Fast and Normal Pipeline (fast_pipeline.exec & normal_pipeline.exec)

Below are the automated bash scripts to run Fast and Normal Pipeline as shown in **Figure 3. & Figure 4.** Scripts developed to run the [Branch 1] nucleotide substitution pipeline was based on the Galaxy Training Project of SARS-CoV-2 Variant Analysis by Beek et al (2021). While [Branch 2] amino acid substitutions pipelines were developed using a combination of available bioinformatics tools. Furthermore, it should be noted that the directory when running the script might differ between local computer and hardware. Therefore, adjustments should be made to ensure the script runs properly.

Installation of Anaconda

Conda refers to an open source package compatible to be run in Windows, Linux, or macOS. Both pipelines would be required to be run in conda environments alongside bioinformatics tools used. To install conda, users may visit the website of anaconda repository (https://repo.anaconda.com/) and select compatible installers within anaconda archive according to their own machine.

Required packages and tools to run the script

Following the conda installation, users may need to test conda environments by simply type 'conda' command inside the terminal. If several lines of text with the version of conda shows up, it indicates the conda environment was successfully installed. Afterwards, users may proceed to install each package required for conda installation. Below listed are the packages and the codes run in this script:

Fastqc → data visualization of fastq (https://www.bioinformatics.babraham.ac.uk/projects/fastqc/)
Trimmomatic → quality control of fastq (http://www.usadellab.org/cms/uploads/supplementary/Trimmomatic/TrimmomaticManual_V0.32.pdf)
BWA → reference assisted assembly alternative (http://bio-bwa.sourceforge.net/bwa.shtml)
Samtools → SAM to BAM conversion, statistics and coverage alignment (http://www.htslib.org/doc/samtools.html)
Bedtools → Convert BAM to fastq (https://bedtools.readthedocs.io/en/latest/)
Bedops → Convert GFF to BED (https://bedops.readthedocs.io/en/latest/)
Seqtk → Convert fastq to fasta (https://github.com/lh3/seqtk)
Seqkit → Sequence analysis and translation to amino acids (https://bioinf.shenwei.me/seqkit/)
Bcftools → Construct consensus sequence using the combination of bam and refseq fasta (http://www.htslib.org/doc/bcftools.html)
Picard → mark duplicate reads (https://broadinstitute.github.io/picard/)
Lofreq → (Viterbi) realign reads to correct misalignments around insertions and deletions, insert indel qualities and call variants (https://csb5.github.io/lofreq/commands/)
SnpEff → annotate variant effects (https://pcingola.github.io/SnpEff/, https://pcingola.github.io/SnpEff/se_running/)
Snpsift → select various effects from the VCF and create a tabular file that is easier to understand for humans. (https://pcingola.github.io/SnpEff/)

```
Conda install -c bioconda fastqc
Conda install -c bioconda trimmomatic
Conda install -c bioconda hisat2
Conda install -c bioconda bwa
```

```
Conda install -c bioconda samtools
Conda install -c bioconda bedtools
Conda install -c bioconda bedops
Conda install -c bioconda seqtk
Conda install -c bioconda seqkit
Conda install -c bioconda bcftools
```
#(bcftools deprecated in base environment, to solve this #problem, install it in new environment; install code → conda #create --name tmp; activate code → conda activate tmp; and #proceed installing bcftools in new environment)

```
Conda install -c bioconda multiqc
Conda install -c bioconda picard
Conda install -c bioconda lofreq
Conda install -c bioconda snpeff
Conda install -c bioconda snpsift
```

Create a new executable bash script

```
(Make executable linux script)
touch fast_pipeline
chmod 774 fast_pipeline
#copy the code of fast pipeline below to the newly created
executable script

touch normal_pipeline
chmod 774 normal_pipeline
#copy the code of normal pipeline below to the newly created
executable script

(add time calculation in each running script)
time [code]
{ time somecodes; } 2>>time.txt
```

(to run the script, head to specified directory where the script is located and type ./scriptname, in this example is ./fast_pipeline)

Automated Bash Script to run Fast Pipeline in All Samples (fast_pipeline.exec)

Before running the script, please ensure the availability of input files including **raw NGS data in FASTQ format, illumina adapter TruSeq3-PE.fa for quality control, SARS-CoV-2 reference genome (NC_045512.2), and SARS-CoV-2 GFF annotation.**

TruSeq3-PE.fa available in github repository of Trimmomatic (https://github.com/timflutre/trimmomatic/blob/master/adapters/TruSeq3-PE.fa)

SARS-CoV-2 reference genome available in NCBI repository (https://www.ncbi.nlm.nih.gov/nuccore/NC_045512)

SARS-CoV-2 GFF annotation available in NCBI repository (https://ftp.ncbi.nlm.nih.gov/genomes/all/GCF/009/858/895/GCF_009858895.2_ASM985889v3/GCF_009858895.2_ASM985889v3_genomic.gff.gz)

After input files check, user may proceed to run the script below

```
#-------------------------FAST PIPELINE SCRIPT-------------------

#batch 1 samples
#SampleB6_S3_L001_R1_001.fastq.gz
#SampleB6_S3_L001_R2_001.fastq.gz
#SampleC5_S1_L001_R1_001.fastq.gz
#SampleC5_S1_L001_R2_001.fastq.gz
#SampleF2_S7_L001_R1_001.fastq.gz
#SampleF2_S7_L001_R2_001.fastq.gz
#SampleF4_S5_L001_R1_001.fastq.gz
#SampleF4_S5_L001_R2_001.fastq.gz

#batch 2 samples
#Sample3_S3_L001_R1_001.fastq.gz
#Sample3_S3_L001_R2_001.fastq.gz
#Sample9_S9_L001_R1_001.fastq.gz
#Sample9_S9_L001_R2_001.fastq.gz
#Sample10_S10_L001_R1_001.fastq.gz
#Sample10_S10_L001_R2_001.fastq.gz
#Sample15_S15_L001_R1_001.fastq.gz
#Sample15_S15_L001_R2_001.fastq.gz

#The list of fastq samples shown above are input files, specify
#input files in quality control below (in this example, we specify
#sample B6), afterwards, let the machine do the work for us!

#(quality control)

fastqc SampleB6_S3_L001_R1_001.fastq.gz
fastqc SampleB6_S3_L001_R2_001.fastq.gz

{ time fastqc SampleB6_S3_L001_R1_001.fastq.gz
SampleB6_S3_L001_R2_001.fastq.gz; } 2>>time.txt
rm SampleF2_S7_L001_R1_001_fastqc.zip
rm SampleF2_S7_L001_R2_001_fastqc.zip

{ time trimmomatic PE -trimlog trim_sample.txt
SampleB6_S3_L001_R1_001.fastq.gz SampleB6_S3_L001_R2_001.fastq.gz
sample_fp.fq.gz sample_fu.fq.gz sample_rp.fq.gz sample_ru.fq.gz
ILLUMINACLIP:TruSeq3-PE.fa:2:30:10 LEADING:3 TRAILING:3
SLIDINGWINDOW:4:15 MINLEN:36; } 2>>time.txt

#(read mapping fast pipeline)
{ time bwa index refseq/bwa_cov_index/NC_045512v2.fasta; }
2>>time.txt
{ time bwa mem refseq/bwa_cov_index/NC_045512v2.fasta
sample_fp.fq.gz sample_rp.fq.gz > sample.sam; } 2>>time.txt

samtools view -Sb sample.sam > sample.bam
samtools sort sample.bam > sample_sort.bam
samtools stats sample_sort.bam > sample_stats.txt
samtools coverage sample_sort.bam > sample_coverage.txt
```

```
#--------------[BRANCH 1] NUCLEOTIDE SUBSTITUTION PIPELINE----------

#(mark duplicates)
PICARD=/home/bi-i3l/anaconda3/share/picard-2.25.2-0/picard.jar

{ time java -jar $PICARD MarkDuplicates REMOVE_DUPLICATES=true
DUPLICATE_SCORING_STRATEGY=SUM_OF_BASE_QUALITIES
OPTICAL_DUPLICATE_PIXEL_DISTANCE=100 VALIDATION_STRINGENCY=LENIENT
use_jdk_deflater=true use_jdk_inflater=true I=sample_sort.bam
O=sample_picard.bam M=sample_picard.txt; } 2>>time.txt

samtools sort sample_picard.bam > sample_picard_sort.bam

#(realign reads)
{ time lofreq viterbi -f refseq/bwa_cov_index/NC_045512v2.fasta -o
sample_realign.bam sample_picard_sort.bam; } 2>>time.txt

#(insert indel qualities)
{ time lofreq indelqual --dindel -f
refseq/bwa_cov_index/NC_045512v2.fasta -o sample_indelqual.bam
sample_picard_sort.bam; } 2>>time.txt

#(variant calling)
{ time lofreq call --verbose --ref
refseq/bwa_cov_index/NC_045512v2.fasta --call-indels --min-cov 50 --
max-depth 1000000 --min-bq 30 --min-alt-bq 30 --min-mq 20 --max-mq
255 --min-jq 0 --min-alt-jq 0 --def-alt-jq 0 --sig 0.01 --bonf
dynamic -o sample.vcf sample_indelqual.bam; } 2>>time.txt

#(variant annotation)
#Depends between GRCh38.86 or GRCh38.99, see java -jar $SNPEFF
#databases | grep -i GRCh38 for more details
SNPEFF=/home/bi-i3l/anaconda3/share/snpeff-4.3.1t-1/snpEff.jar
CONFIG=/home/bi-i3l/anaconda3/pkgs/snpeff-4.3.1t-1/share/snpeff-
4.3.1t-1/snpEff.config

{ time java -Xmx8g -jar $SNPEFF -c $CONFIG -v GRCh38.86 sample.vcf >
sample_annotated.vcf; } 2>>time.txt

#(variant extraction to csv)
SNPSIFT=/home/bi-i3l/anaconda3/share/snpsift-4.3.1t-1/SnpSift.jar

{ time java -jar $SNPSIFT extractFields -s "," -e "."
sample_annotated.vcf CHROM POS ID REF ALT QUAL DP AF SB DP4 >
sample_annotated.csv; } 2>>time.txt

#-------------[BRANCH 2] AMINO ACIDS SUBSTITUTION PIPELINE----------

#(construct consensus sequences from bam file)

{ time samtools mpileup -uf refseq/bwa_cov_index/NC_045512v2.fasta
sample_sort.bam | bcftools call -c | vcfutils.pl vcf2fq >
sample_cns.fq; } 2>>time_consensus.txt

#(filter good quality bases, convert fastq to fasta)
```

```
{ seqtk seq -aQ64 -q20 -nN sample_cns.fq > sample_cns.fasta; }
2>>time_consensus.txt
```

#NC_045512v2(using gff of sars-cov-2, convert gff to bed, mapping bed to corresponding region in fasta)

```
{ time sortBed -i NC_045512v2.gff | gff2bed > NC_045512v2.bed; }
2>>time_consensus.txt

{ time bedtools getfasta -fi refseq/bwa_cov_index/NC_045512v2.fasta
-bed NC_045512v2.bed -name > NC_045512v2_bed.fasta; }
2>>time_consensus.txt

{ time seqkit split -i NC_045512v2_bed.fasta; }
2>>time_consensus.txt
```

#SAMPLES(using gff of sars-cov-2, convert gff to bed, mapping bed to corresponding region in fasta)

```
{ time bedtools getfasta -fi sample_cns.fasta -bed NC_045512v2.bed -
name > sample_bed.fasta; } 2>>time_consensus.txt
```

#(split fasta according to sequences)

```
{ time seqkit split -i sample_bed.fasta; } 2>>time_consensus.txt
```

#(clean excess files, optional and user may adjust according to their own needs)

```
rm sample_fu.fq.gz sample_ru.fq.gz sample_indelqual.bam
sample_picard_sort.bam sample_picard.bam sample_realign.bam
sample_sort.bam sample.sam snpEff_genes.txt sample_human.sam
sample_human.bam sample_human_sort.bam sample_unmapped.bam
sample_unmapped_sort.bam sample_fp_unmapped.fastq
sample_rp_unmapped.fastq
```

Automated Bash Script to run Normal Pipeline in All Samples (normal_pipeline.exec)

Before running the script, please ensure the availability of input files including **raw NGS data in FASTQ format, illumina adapter TruSeq3-PE.fa for quality control, SARS-CoV-2 reference genome (NC_045512.2), and SARS-CoV-2 GFF annotation.**

TruSeq3-PE.fa available in github repository of Trimmomatic (https://github.com/timflutre/trimmomatic/blob/master/adapters/TruSeq3-PE.fa)

SARS-CoV-2 reference genome available in NCBI repository (https://www.ncbi.nlm.nih.gov/nuccore/NC_045512)

SARS-CoV-2 GFF annotation available in NCBI repository (https://ftp.ncbi.nlm.nih.gov/genomes/all/GCF/009/858/895/GCF_009858895.2_ASM985889v3/GCF_009858895.2_ASM985889v3_genomic.gff.gz)

After input files check, user may proceed to run the script below

```
#------------------------NORMAL PIPELINE SCRIPT--------------------

#batch 1 samples
#SampleB6_S3_L001_R1_001.fastq.gz
#SampleB6_S3_L001_R2_001.fastq.gz
#SampleC5_S1_L001_R1_001.fastq.gz
#SampleC5_S1_L001_R2_001.fastq.gz
#SampleF2_S7_L001_R1_001.fastq.gz
#SampleF2_S7_L001_R2_001.fastq.gz
#SampleF4_S5_L001_R1_001.fastq.gz
#SampleF4_S5_L001_R2_001.fastq.gz

#batch 2 samples
#Sample3_S3_L001_R1_001.fastq.gz
#Sample3_S3_L001_R2_001.fastq.gz
#Sample9_S9_L001_R1_001.fastq.gz
#Sample9_S9_L001_R2_001.fastq.gz
#Sample10_S10_L001_R1_001.fastq.gz
#Sample10_S10_L001_R2_001.fastq.gz
#Sample15_S15_L001_R1_001.fastq.gz
#Sample15_S15_L001_R2_001.fastq.gz

#The list of fastq samples shown above are input files, specify
#input files in quality control below (in this example, we specify
#sample B6), afterwards, let the machine do the work for us!

#(quality control)

fastqc SampleB6_S3_L001_R1_001.fastq.gz
fastqc SampleB6_S3_L001_R2_001.fastq.gz

{ time fastqc SampleB6_S3_L001_R1_001.fastq.gz
SampleB6_S3_L001_R2_001.fastq.gz; } 2>>time.txt
rm SampleF2_S7_L001_R1_001_fastqc.zip
rm SampleF2_S7_L001_R2_001_fastqc.zip
```

```
{ time trimmomatic PE -trimlog trim_sample.txt
SampleB6_S3_L001_R1_001.fastq.gz SampleB6_S3_L001_R2_001.fastq.gz
sample_fp.fq.gz sample_fu.fq.gz sample_rp.fq.gz sample_ru.fq.gz
ILLUMINACLIP:TruSeq3-PE.fa:2:30:10 LEADING:3 TRAILING:3
SLIDINGWINDOW:4:15 MINLEN:36; } 2>>time.txt

#(read mapping normal pipeline)
{ time bwa index refseq/bwa_human_index/GRCh38.fna; } 2>>time.txt
{ time bwa mem refseq/bwa_human_index/GRCh38.fna sample_fp.fq.gz
sample_rp.fq.gz > sample_human.sam; } 2>>time.txt

samtools view -Sb sample_human.sam > sample_human.bam
samtools sort sample_human.bam > sample_human_sort.bam
samtools view -bf4 sample_human_sort.bam > sample_unmapped.bam
samtools sort -n sample_unmapped.bam > sample_unmapped_sort.bam

bedtools bamtofastq -i sample_unmapped_sort.bam -fq
sample_fp_unmapped.fastq -fq2 sample_rp_unmapped.fastq

{ time bwa mem refseq/bwa_cov_index/NC_045512v2.fasta
sample_fp_unmapped.fastq sample_rp_unmapped.fastq > sample.sam; }
2>>time.txt

samtools view -Sb sample.sam > sample.bam
samtools sort sample.bam > sample_sort.bam
samtools stats sample_sort.bam > sample_stats.txt
samtools coverage sample_sort.bam > sample_coverage.txt

#--------------[BRANCH 1] NUCLEOTIDE SUBSTITUTION PIPELINE----------

#(mark duplicates)
PICARD=/home/bi-i3l/anaconda3/share/picard-2.25.2-0/picard.jar

{ time java -jar $PICARD MarkDuplicates REMOVE_DUPLICATES=true
DUPLICATE_SCORING_STRATEGY=SUM_OF_BASE_QUALITIES
OPTICAL_DUPLICATE_PIXEL_DISTANCE=100 VALIDATION_STRINGENCY=LENIENT
use_jdk_deflater=true use_jdk_inflater=true I=sample_sort.bam
O=sample_picard.bam M=sample_picard.txt; } 2>>time.txt

samtools sort sample_picard.bam > sample_picard_sort.bam

#(realign reads)
{ time lofreq viterbi -f refseq/bwa_cov_index/NC_045512v2.fasta -o
sample_realign.bam sample_picard_sort.bam; } 2>>time.txt

#(insert indel qualities)
{ time lofreq indelqual --dindel -f
refseq/bwa_cov_index/NC_045512v2.fasta -o sample_indelqual.bam
sample_picard_sort.bam; } 2>>time.txt

#(variant calling)
{ time lofreq call --verbose --ref
refseq/bwa_cov_index/NC_045512v2.fasta --call-indels --min-cov 50 --
max-depth 1000000 --min-bq 30 --min-alt-bq 30 --min-mq 20 --max-mq
255 --min-jq 0 --min-alt-jq 0 --def-alt-jq 0 --sig 0.01 --bonf
dynamic -o sample.vcf sample_indelqual.bam; } 2>>time.txt
```

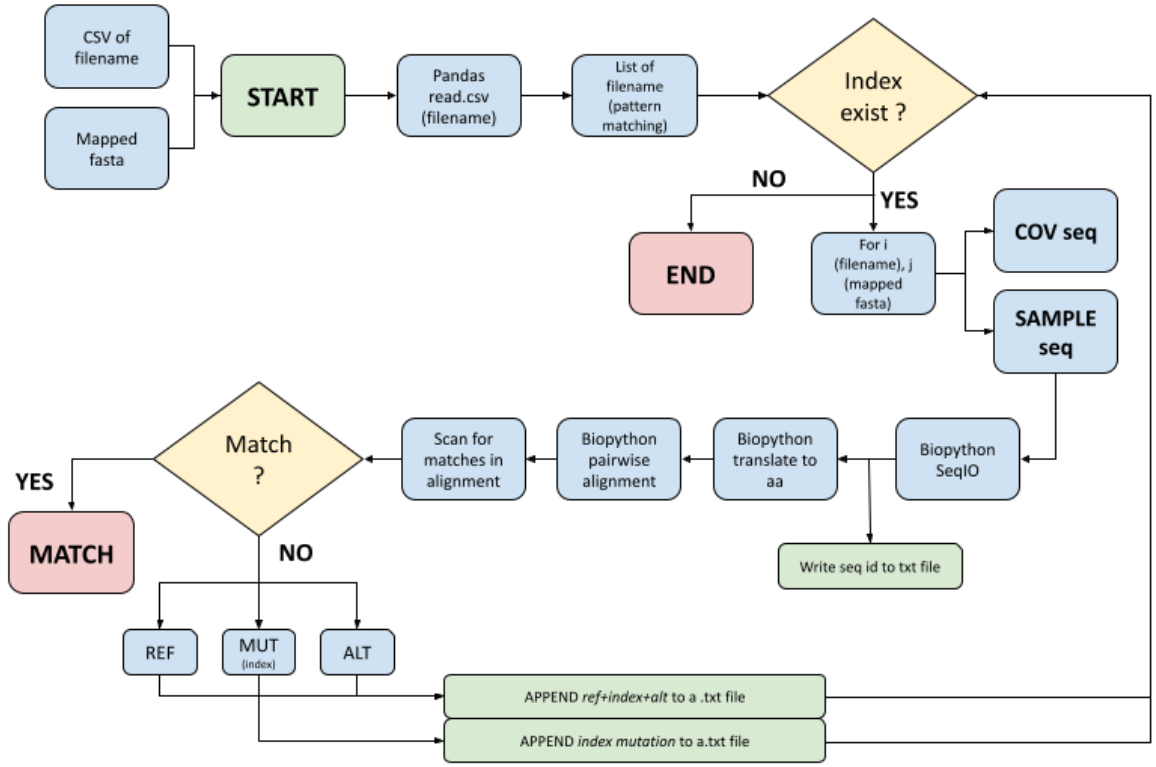
```
#(variant annotation)
#Depends between GRCh38.86 or GRCh38.99, see java -jar $SNPEFF
#databases | grep -i GRCh38 for more details
SNPEFF=/home/bi-i3l/anaconda3/share/snpeff-4.3.1t-1/snpEff.jar
CONFIG=/home/bi-i3l/anaconda3/pkgs/snpeff-4.3.1t-1/share/snpeff-
4.3.1t-1/snpEff.config

{ time java -Xmx8g -jar $SNPEFF -c $CONFIG -v GRCh38.86 sample.vcf >
sample_annotated.vcf; } 2>>time.txt

#(variant extraction to csv)
SNPSIFT=/home/bi-i3l/anaconda3/share/snpsift-4.3.1t-1/SnpSift.jar

{ time java -jar $SNPSIFT extractFields -s "," -e "."
sample_annotated.vcf CHROM POS ID REF ALT QUAL DP AF SB DP4 >
sample_annotated.csv; } 2>>time.txt

#-------------[BRANCH 2] AMINO ACIDS SUBSTITUTION PIPELINE----------

#(construct consensus sequences from bam file)

{ time samtools mpileup -uf refseq/bwa_cov_index/NC_045512v2.fasta
sample_sort.bam | bcftools call -c | vcfutils.pl vcf2fq >
sample_cns.fq; } 2>>time_consensus.txt

#(filter good quality bases, convert fastq to fasta)

{ seqtk seq -aQ64 -q20 -nN sample_cns.fq > sample_cns.fasta; }
2>>time_consensus.txt

#NC_045512v2(using gff of sars-cov-2, convert gff to bed, mapping
bed to corresponding region in fasta)

{ time sortBed -i NC_045512v2.gff | gff2bed > NC_045512v2.bed; }
2>>time_consensus.txt

{ time bedtools getfasta -fi refseq/bwa_cov_index/NC_045512v2.fasta
-bed NC_045512v2.bed -name > NC_045512v2_bed.fasta; }
2>>time_consensus.txt

{ time seqkit split -i NC_045512v2_bed.fasta; }
2>>time_consensus.txt

#SAMPLES(using gff of sars-cov-2, convert gff to bed, mapping bed to
corresponding region in fasta)

{ time bedtools getfasta -fi sample_cns.fasta -bed NC_045512v2.bed -
name > sample_bed.fasta; } 2>>time_consensus.txt

#(split fasta according to sequences)

{ time seqkit split -i sample_bed.fasta; } 2>>time_consensus.txt

#(clean excess files, optional and user may adjust according to
```

their own needs)

```
rm sample_fu.fq.gz sample_ru.fq.gz sample_indelqual.bam
sample_picard_sort.bam sample_picard.bam sample_realign.bam
sample_sort.bam sample.sam snpEff_genes.txt sample_human.sam
sample_human.bam sample_human_sort.bam sample_unmapped.bam
sample_unmapped_sort.bam sample_fp_unmapped.fastq
sample_rp_unmapped.fastq
```

Python Algorithm for Pairwise Alignment and Detection of Amino Acid Mutations (pairwise_alignment_to_txt.py and **txt_to_excel_workbook.py)**

[Branch 2] amino acid substitution pipeline would result in mapped consensus to SARS-CoV-2 bed corresponding to each region inside SARS-CoV-2 genome. These mapped fasta were subjected for pairwise alignment to detect subsequent mutations. Mutations will be appended to a .txt file with a format similar to .csv. However, owing to the large number of fasta files after mapping to region (1 original consensus sequence are mapped to 59 region, resulting in 59 fasta files in each sample), as pattern matching method was used in this algorithm, a .csv file containing the filename will be required to act as the 'bait' to capture the mapped fasta file. Afterwards, biopython will detect input files, translate fasta to amino acids, and pairwise alignments will be conducted. The algorithm will scan for mutation and if it manages to detect one, reference nucleotides alongside alternate nucleotides and position will be appended to a text file. Iteration will be done thoroughly until all samples are fully covered and mutations were fully detected. **Appendices 1.** summarizes the whole workflow in pairwise_alignment_to_txt.py.

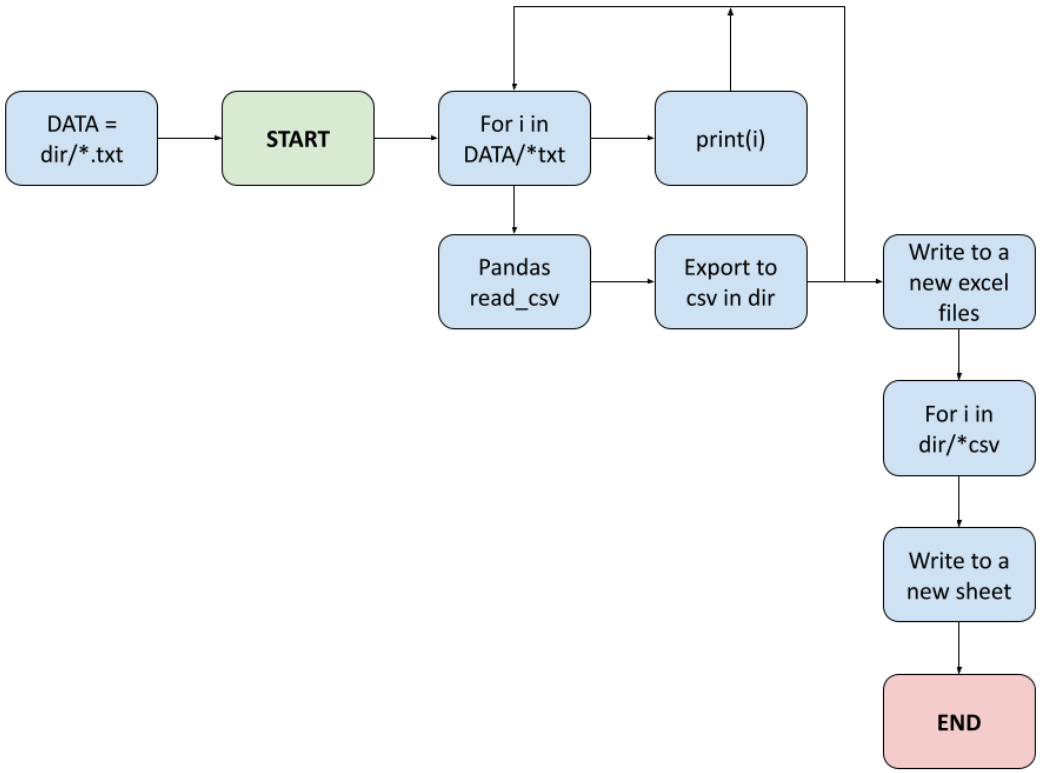


Appendices 1. Workflow illustration for pairwise_alignment_to_txt.py

```
from Bio import SeqIO
from Bio import pairwise2
import glob
import pandas as pd

#-----------------------------------------------------------------

full_data = []
not_full_data = []

#samples were categorized into 2 based on their length, here we have
#full (29903 and notfull (29834 - 29903). The categorization was
#applied due to #the length of sequences, if sequences not met
#29903, subsequently, modification to mapped #NC_045512.2 should be
#applied otherwise will result in error)

#here, 2 csv file were created as the basis for pattern matching, it
was applied owing to #long naming of samples from UGM

full = pd.read_csv('fasta_stats_full.csv', sep=',')
not_full = pd.read_csv('fasta_stats_not_full.csv', sep=',')

for i in full['file']:
    full_data.append('./BATCH_UGM/'+i+'/*.fasta')
#for i in not_full['file']:
#    not_full_data.append('./BATCH_UGM/'+i+'/*.fasta')

for i, j in zip(full['file'], full_data):
    COV =
list(sorted(glob.glob("./NC_045512v2_bed.fasta.split_full/*.fasta"))
)
    SAMPLE = list(sorted(glob.glob(j)))

    filename = i+'.txt'

    print('FILE OUTPUT: '+filename)

    for x, y in zip(COV, SAMPLE):
        seq1 = SeqIO.read(x, "fasta")
        seq2 = SeqIO.read(y, "fasta")
        print(seq1)
        print('\n')
        print(seq2)
        print('\n')

        for seq_record in SeqIO.parse(x, "fasta"):
            record_1 = seq_record.seq

        for seq_record in SeqIO.parse(y, "fasta"):
            with open(filename, 'a') as f:
                f.writelines(seq_record.id)
            record_2 = seq_record.seq

        pro1 = record_1.translate()
        pro2 = record_2.translate()
      #-----------------------------------------------------------------
```

```
------
        alignments = pairwise2.align.localms(pro1, pro2, 2, -1, -1,
-1)

        match = []
        ref = []
        alt = []

        c = 1
        mut = []
        for a, b in zip(alignments[0][0],alignments[0][1]):
                if a == b:
                        match.append('|')
                        c += 1
                else:
                        match.append('*')
                        ref.append(a)
                        alt.append(b)
                        mut.append(c)
                        c += 1

        m="".join(match)
        s=[]
        s.append(alignments[0][0]+'\n')
        s.append(m+'\n')
        s.append(alignments[0][1])

        alignedSeqs="".join(s)
        print(alignedSeqs)
        print('\n')

        for (q,w,r) in zip(ref, alt, mut):
            with open(filename, 'a') as f:
                f.writelines(';'+str(q)+ str(r) + str(w))

        with open(filename, 'a') as f:
            f.writelines(';' + str(mut))
            f.writelines('\n')
```

Text files generated from previous python code will be converted to csv and compiled into one excel workbook file containing the amino acids mutations in each sample. **Appendices 2.** summarizes the whole workflow in **txt_to_excel_workbook.py**.



Appendices 2. Workflow illustration for txt_to_excel_workbook.py

```
import pandas as pd
import sys
import os
import glob
from pathlib import Path

#this code will write subsequent alignment result in the form of
excel workbook
#first, we need to convert all the text files from previous
procedure to a new csv file

DATA = list(sorted(glob.glob("./ugm_covid_full_length/*.txt")))
root = 'ugm_covid_csv'

for i in DATA:
    print(i)
    data = pd.read_csv(i, sep=';')
    data.to_csv(os.path.join(root, i[71:-28]+'.csv'), sep=';',
index=False)

# COMBINE ALL CSV TO ONE EXCEL FILE

extension = 'csv'
all_filenames = [i for i in glob.glob("./ugm_covid_csv/*.csv")]
```

```
writer = pd.ExcelWriter('UGM_COVID_36_FULL.xlsx')  # Arbitrary
output name
for i in all_filenames:
    txt = Path(i).read_text()
    txt = txt.replace(',', '.')

    text_file = open(i, "w")
    text_file.write(txt)
    text_file.close()

    print(i[16:])
    df = pd.read_csv(i, sep=';', encoding='utf-8')
    df.to_excel(writer, sheet_name=os.path.splitext(i[16:])[0],
index=False)
writer.save()
```

RUNTIME EXECUTION OF FAST PIPELINE IN BATCH 1 AND BATCH 2 SAMPLES

Step	Tools	Input Files	Output Files	TIME (S)			
				B6 Fast Pipeline	C5 Fast Pipeline	F2 Fast Pipeline	F4 Fast Pipeline
Quality control	FASTQC	FASTQ	HTML	48.1	14.2	13.1	9.1
	TRIMMOMATIC	FASTQ	FASTQ	210	50.6	46.6	25.8
Indexing reference genome (SARS-CoV-2)	BWA-MEM	FASTA	INDEX FILES	0.3	0.3	0.3	0.3
Read mapping to reference genome (SARS-CoV-2)	BWA-MEM	FASTQ (Samples) and INDEX FILES	BAM	154	40.5	55.5	25.2
Post-read mapping processing	SAMTOOLS	BAM	BAM	42	6.8	5.6	3.3
	PICARD	BAM	BAM	59.3	17	15.4	9.5
Realign reads	LOFREQ	BAM & FASTA (Reference)	BAM	17.3	7.2	6	3.5
Insert indel qualities	LOFREQ	BAM & FASTA (Reference)	BAM	12.6	5.1	5.9	3.2
Variant calling	LOFREQ	BAM & FASTA (Reference)	VCF	494	149	6	20.5
Variant annotation	SNPEFF	VCF	VCF	35.9	36.2	36.6	36.6
Variant extraction to csv	SNPSIFT	VCF	CSV	0.3	0.2	0.2	0.18
Total Time Required [Branch 1]				**1073.8**	**327.1**	**191.2**	**137.18**

Appendices 3. Runtime execution of Batch 1 samples in detecting nucleotide substitution (Branch 1 - Fast Pipeline).

Step	Tools	Input Files	Output Files	TIME (S)			
				S3 Fast Pipeline	S9 Fast Pipeline	S10 Fast Pipeline	S15 Fast Pipeline
Quality control	FASTQC	FASTQ	HTML	82	36.7	15.1	28.1
	TRIMMOMATIC	FASTQ	FASTQ	378	148	53.8	116
Indexing reference genome (SARS-CoV-2)	BWA-MEM	FASTA	INDEX FILES	0.04	0.03	0.03	0.03
Read mapping to reference genome (SARS-CoV-2)	BWA-MEM	FASTQ (Samples) and INDEX FILES	BAM	248	125	64	111
Post-read mapping processing	SAMTOOLS	BAM	BAM	86	36.9	7.9	24.2
	PICARD	BAM	BAM	107	51.1	18.5	39.9
Realign reads	LOFREQ	BAM & FASTA (Reference)	BAM	21.1	19.8	6.7	16.2
Insert indel qualities	LOFREQ	BAM & FASTA (Reference)	BAM	18	18.7	6.1	14.2
Variant calling	LOFREQ	BAM & FASTA (Reference)	VCF	854	224	96	169
Variant annotation	SNPEFF	VCF	VCF	37.2	36.7	36.5	36.9
Variant extraction to csv	SNPSIFT	VCF	CSV	0.3	0.2	0.02	0.02
Total Time Required [Branch 1]				**1831.64**	**697.13**	**304.65**	**555.55**

Appendices 4. Runtime execution of Batch 2 samples in detecting nucleotide substitution (Branch 1 - Fast Pipeline).

Step	Tools	Input Files	Output Files	TIME (S)			
				B6 Fast Pipeline	C5 Fast Pipeline	F2 Fast Pipeline	F4 Fast Pipeline
Convert GFF SARS-CoV-2 to BED	BEDTOOLS	GFF	BED	0.01	0.01	0.01	0.01
Map SARS-CoV-2 genome to SARS-CoV-2 BED	BEDTOOLS	BED & FASTA	FASTA	0.01	0.01	0.01	0.01
Split fasta file	SEQKIT	FASTA	FASTA	0.01	0.01	0.01	0.01
Construct consensus sequence of samples	SAMTOOLS	BAM	FASTQ	574	117	6.4	13.7
	BCFTOOLS						
Obtain good quality bases and convert to fasta	SEQTK	FASTQ	FASTA	0.01	0.01	0.01	0.01
Map fasta to corresponding SARS-CoV-2 BED file	BEDTOOLS	FASTA (samples) & BED (Reference)	FASTA	0.01	0.01	0.01	0.01
Split fasta file	SEQKIT	FASTA	FASTA	0.01	0.01	0.01	0.01
Pairwise Alignment	PYTHON (BIOPYTHON)	FASTA	TXT	130.1	130.1	126.8	135.5
Total Time Required [Branch 2]				**704.16**	**247.16**	**133.26**	**149.26**

Appendices 5. Runtime execution of Batch 1 samples in detecting amino acids substitution (Branch 2 - Fast Pipeline).

Step	Tools	Input Files	Output Files	TIME (S)			
				S3 Fast Pipeline	S9 Fast Pipeline	S10 Fast Pipeline	S15 Fast Pipeline
Download GFF Annotation of SARS-CoV-2	-	GFF	GFF	-	-	-	-
Convert GFF SARS-CoV-2 to BED	BEDTOOLS	GFF	BED	13.7	13.7	13.7	13.7

Map SARS-CoV-2 genome to SARS-CoV-2 BED	BEDTOOLS	BED & FASTA	FASTA	0.01	0.01	0.01	0.01
Split fasta file	SEQKIT	FASTA	FASTA	0.01	0.01	0.01	0.01
Construct consensus sequence of samples	SAMTOOLS	BAM	FASTQ	1083	200	87	145
	BCFTOOLS						
Obtain good quality bases and convert to fasta	SEQTK	FASTQ	FASTA	0.01	0.01	0.01	0.01
Map fasta to corresponding SARS-CoV-2 BED file	BEDTOOLS	FASTA (samples) & BED (Reference)	FASTA	0.01	0.01	0.01	0.01
Split fasta file	SEQKIT	FASTA	FASTA	0.01	0.01	0.01	0.01
Pairwise Alignment	PYTHON (BIOPYTHON)	FASTA	TXT	131.7	125.9	132.1	134.6
Total Time Required [Branch 2]				**1228.45**	**339.65**	**232.85**	**293.35**

Appendices 6. Runtime execution of Batch 2 samples in detecting amino acids substitution (Branch 2 - Fast Pipeline).

RUNTIME EXECUTION OF NORMAL PIPELINE IN BATCH 1 AND BATCH 2 SAMPLES

Step	Tools	Input Files	Output Files	TIME (S)			
				B6 Normal Pipeline	C5 Normal Pipeline	F2 Normal Pipeline	F4 Normal Pipeline
Quality control	FASTQC	FASTQ	HTML	50.6	15.6	16	11
	TRIMMOMATIC	FASTQ	FASTQ	3.36	51.4	45.7	26.2
Indexing reference genome (GRCh38.86)	BWA-MEM	FASTA	INDEX FILES	3082	3201	3423	3153
Read mapping to reference genome (GRCh38.86)	BWA-MEM	FASTQ (Samples) and INDEX FILES	BAM	1976	313	360	207

Obtain unmapped reads	SAMTOOLS	BAM	BAM	0.01	0.01	0.01	0.01
Convert unmapped reads back to fastq	BEDTOOLS	BAM	FASTQ	0.01	0.01	0.01	0.01
Indexing reference genome (SARS-CoV-2)	BWA-MEM	FASTA	INDEX FILES	0.3	0.3	0.3	0.3
Read mapping to reference genome (SARS-CoV-2)	BWA-MEM	FASTQ (Samples) and INDEX FILES	BAM	124	24.2	0.8	2.2
Post-read mapping processing	SAMTOOLS	BAM	BAM	31.4	4.1	0.17	0.4
	PICARD	BAM	BAM	46.2	11.5	3.3	3.3
Realign reads	LOFREQ	BAM & FASTA (Reference)	BAM	11.8	4.5	0.2	0.5
Insert indel qualities	LOFREQ	BAM & FASTA (Reference)	BAM	7.7	2.7	0.1	0.4
Variant calling	LOFREQ	BAM & FASTA (Reference)	VCF	7.3	138	5.2	19
Variant annotation	SNPEFF	VCF	VCF	36.7	38.9	0.4	37
Variant extraction to csv	SNPSIFT	VCF	CSV	0.3	0.3	0.2	0.2
Total Time Required [Branch 1]				**5377.68**	**3805.52**	**3855.39**	**3460.52**

Appendices 7. Runtime execution of Batch 1 samples in detecting nucleotide substitution (Branch 1 - Normal Pipeline).

Step	Tools	Input Files	Output Files	TIME (S)			
				S3 Normal Pipeline	S9 Normal Pipeline	S10 Normal Pipeline	S15 Normal Pipeline
Quality control	FASTQC	FASTQ	HTML	89	37.9	17.1	30.7
	TRIMMOMATIC	FASTQ	FASTQ	362	148	53.1	116
Indexing reference genome (GRCh38.86)	BWA-MEM	FASTA	INDEX FILES	3160	3158	3103	3088
Read mapping to	BWA-MEM	FASTQ (Samples)	BAM	722	852	266	638

reference genome (GRCh38.86)		and INDEX FILES					
Obtain unmapped reads	SAMTOOLS	BAM	BAM	0.01	0.01	0.01	0.01
Convert unmapped reads back to fastq	BEDTOOLS	BAM	FASTQ	0.01	0.01	0.01	0.01
Indexing reference genome (SARS-CoV-2)	BWA-MEM	FASTA	INDEX FILES	0.03	0.03	0.03	0.03
Read mapping to reference genome (SARS-CoV-2)	BWA-MEM	FASTQ (Samples) and INDEX FILES	BAM	220	37.2	16.7	57.7
Post-read mapping processing	SAMTOOLS	BAM	BAM	76	8.3	3.5	5.7
	PICARD	BAM	BAM	86	18.4	8.5	12.3
Realign reads	LOFREQ	BAM & FASTA (Reference)	BAM	17.5	4.5	2.2	4
Insert indel qualities	LOFREQ	BAM & FASTA (Reference)	BAM	14.6	4	1.7	2.9
Variant calling	LOFREQ	BAM & FASTA (Reference)	VCF	722	195	82	154
Variant annotation	SNPEFF	VCF	VCF	37.7	37.3	37	36.9
Variant extraction to csv	SNPSIFT	VCF	CSV	0.3	0.03	0.02	0.04
Total Time Required [Branch 1]				**5507.15**	**4500.68**	**3590.87**	**4146.29**

Appendices 8. Runtime execution of Batch 2 samples in detecting nucleotide substitution (Branch 1 - Normal Pipeline).

Step	Tools	Input Files	Output Files	TIME (S)			
				B6 Normal Pipeline	C5 Normal Pipeline	F2 Normal Pipeline	F4 Normal Pipeline
Convert GFF SARS-CoV-2 to BED	BEDTOOLS	GFF	BED	0.01	0.01	0.01	0.01
Map SARS-CoV-2 genome to SARS-CoV-2 BED	BEDTOOLS	BED & FASTA	FASTA	0.01	0.01	0.01	0.01
Split fasta file	SEQKIT	FASTA	FASTA	0.01	0.01	0.01	0.01
Construct consensus sequence of samples	SAMTOOLS	BAM	FASTQ	549	108	4.5	13.1
	BCFTOOLS						
Obtain good quality bases and convert to fasta	SEQTK	FASTQ	FASTA	0.01	0.01	0.01	0.01
Map fasta to corresponding SARS-CoV-2 BED file	BEDTOOLS	FASTA (samples) & BED (Reference)	FASTA	0.01	0.01	0.01	0.01
Split fasta file	SEQKIT	FASTA	FASTA	0.01	0.01	0.01	0.01
Pairwise Alignment	PYTHON (BIOPYTHON)	FASTA	TXT	64.6	66.9	64.1	65.8
Total Time Required [Branch 2]				**613.66**	**174.96**	**68.66**	**78.96**

Appendices 9. Runtime execution of Batch 1 samples in detecting amino acids substitution (Branch 2 - Normal Pipeline).

Step	Tools	Input Files	Output Files	TIME (S)			
				S3 Normal Pipeline	S9 Normal Pipeline	S10 Normal Pipeline	S15 Normal Pipeline
Download GFF Annotation of SARS-CoV-2	-	GFF	GFF	-	-	-	-
Convert GFF SARS-CoV-2 to BED	BEDTOOLS	GFF	BED	13.7	13.7	13.7	13.7

Map SARS-CoV-2 genome to SARS-CoV-2 BED	BEDTOOLS	BED & FASTA	FASTA	0.01	0.01	0.01	0.01
Split fasta file	SEQKIT	FASTA	FASTA	0.01	0.01	0.01	0.01
Construct consensus sequence of samples	SAMTOOLS	BAM	FASTQ	935	177	79	134
	BCFTOOLS						
Obtain good quality bases and convert to fasta	SEQTK	FASTQ	FASTA	0.01	0.01	0.01	0.01
Map fasta to corresponding SARS-CoV-2 BED file	BEDTOOLS	FASTA (samples) & BED (Reference)	FASTA	0.01	0.01	0.01	0.01
Split fasta file	SEQKIT	FASTA	FASTA	0.01	0.01	0.01	0.01
Pairwise Alignment	PYTHON (BIOPYTHON)	FASTA	TXT	65.05	64.1	63.7	62.8
Total Time Required [Branch 2]				**1013.8**	**254.85**	**156.45**	**210.55**

Appendices 10. Runtime execution of Batch 2 samples in detecting amino acids substitution (Branch 2 - Normal Pipeline).