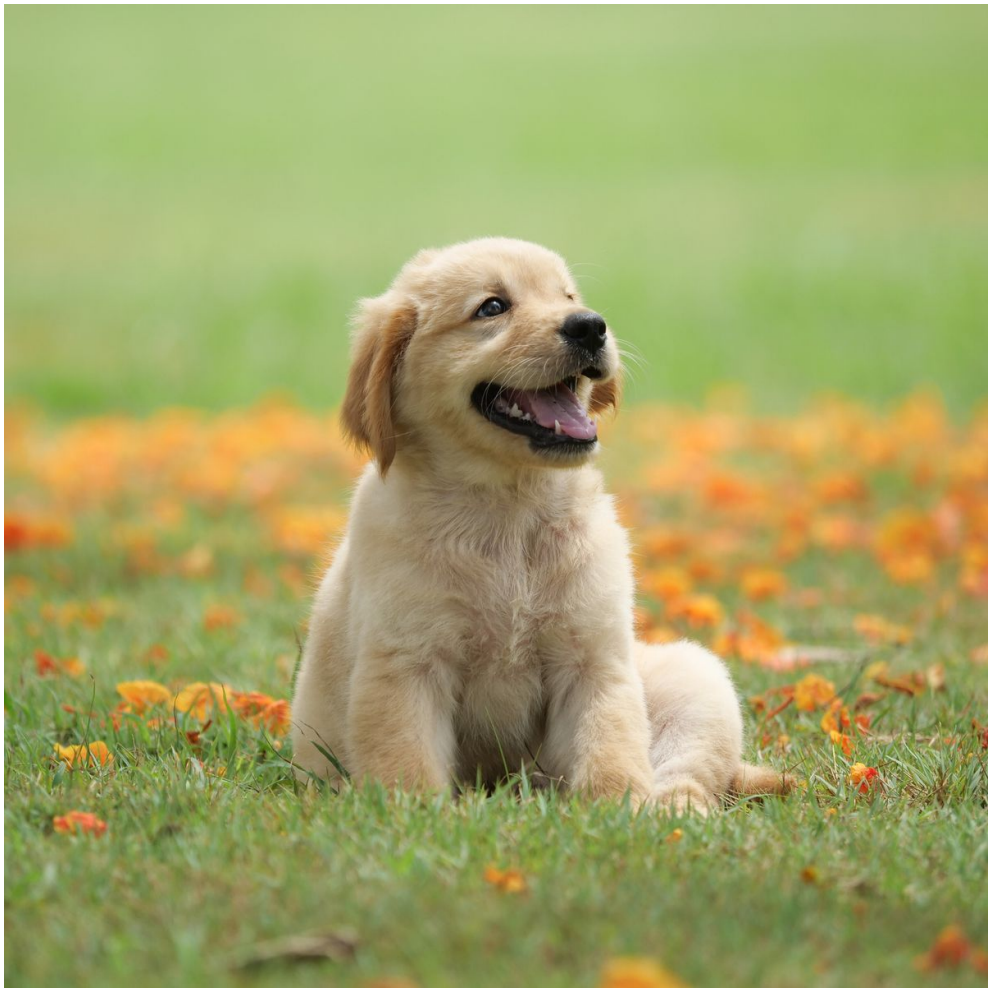
Université Libre de Bruxelles & Vrije Universiteit Brussel

DeepDoggo

LACCHINI Noah, ADIB Ayoub



June 2023

Contents

1 Introduction

1.1 The viral world

The biological world is really complex, it is not always the biggest or the smartest that always win. Sometimes, the smallest entities can be our undoing. One of them is so small that it can't be seen by any of us or by other living beings, and isn't classified as a living being. They are capable of causing disease, destroying cells and sometimes killing; these complex things are called viruses.

They are found as particles in a variety of environments, their role being to act as infectious agents. The viral architecture is really simple, consisting of a protein box (capsid) enveloping the genetic material (DNA or RNA). The capsid protects the genetic material from its environment. Viruses are obligate intracellular parasites, needing a host to replicate and propagate. They are capable of infecting a wide range of organisms, including humans, animals, plants and even bacterias [JJ17, MA16]. The protocol is simple: the virus infects a host cell and injects its genetic material. The viral genetic material can then be replicated and used to produce proteins. Once the genetic material has been replicated and all the proteins synthesized, the virus can leave the cell environment and acquire its infectious characteristics.

Their ecological role is varied, acting on a broad spectrum and present in all environments: in water, in the air we breathe and even in the soil. They regulate animal populations by infecting the most abundant, preventing epidemics and maintaining equilibrium. Since they use the host's cellular machinery to replicate, they can transfer genetic material between different organisms. This phenomenon, known as "horizontal gene transfer", can lead to the acquisition of new genetic traits and contribute to the genetic diversity and evolutionary potential of ecosystems.

Although they have many positive ecological aspects, it's important to recognize that they also have detrimental effects on the ecosystem and the global economy. Recently, humanity experienced this with the events of the Covid-19 epidemic, which showed us more than ever how damaging they can be [NS23]. Viruses can cause a variety of illnesses in their hosts, ranging from mild infections like the common cold to more serious diseases such as influenza, Ebola or COVID-19. They can spread from person to person through direct contact, respiratory droplets or contaminated surfaces. [EYS+23]

1.2 Oncoviruse

Certain type of virus have the ability to cause cancer in animals, including humans, they are called oncoviruses. These viruses can directly contribute to the development of cancer by altering the genetic material (DNA or RNA) of host cells, leading to uncontrolled cell growth and the formation of tumors. Around 15% of human cancer cases are attributed to viral infections [EYS+23, AN17]. Oncoviruses can infect various cell types in the body, including cells of the immune system, epithelial cells, and other tissues. They can either integrate their genetic material into the host cell's DNA or persist in the host cell as an episome (a separate genetic element).

Some well-known examples of oncoviruses :

- The human papillomavirus (HPV), is known to cause cervical cancer. Persistent infection with high-risk HPV strains, such as HPV-16 and HPV-18, is a major risk factor for cervical cancer. These viruses can integrate their DNA into the host cell's genome, leading to the dysregulation of cell cycle control and promoting the progression of cervical cells toward malignancy [zH09];
- The hepatitis B Virus (HBV) and Hepatocellular Carcinoma (HCC) are known to contributing to the development of liver cancer. Chronic HBV infection is strongly associated with the development of HCC. The viral proteins and integration of viral DNA into the host genome can cause chronic inflammation, DNA damage, and genomic instability. [Che06];
- Hepatitis C Virus (HCV) and Hepatocellular Carcinoma (HCC) promotes liver cancer development through direct viral cytotoxicity, chronic inflammation, and the activation of cellular pathways involved in cell proliferation and survival [ES12];

- Epstein-Barr Virus (EBV) and Burkitt Lymphoma, Nasopharyngeal Carcinoma. EBV is associated with several malignancies, including Burkitt lymphoma, a B-cell lymphoma commonly found in children, and nasopharyngeal carcinoma, a cancer of the nasopharynx. It infects and transforms B cells, leading to uncontrolled cell growth and tumor formation [You04];
- Human T-cell Lymphotropic Virus-1 (HTLV-1) and Adult T-cell Leukemia/Lymphoma (ATLL). HTLV-1 is linked to the development of ATLL, an aggressive T-cell malignancy. The virus integrates into the host genome, perturbing cellular gene expression and immune responses, ultimately leading to the transformation of T cells and the development of ATLL [Ban04].

These are just a few examples of well-established virus-cancer associations. The mechanisms by which viruses contribute to cancer development are complex and multifaceted, involving viral onco-proteins, chronic inflammation, immune evasion, and disruption of cellular regulatory pathways.

1.3 Deeplearning

Recent developments in next-generation sequencing have made it possible to examine human cancer databases such as The Cancer Genome Atlas (TCGA) to identify viral sequences in cancer tissues [EYS+23, Sal13]. Other methods have been tested, such as RNA sequencing to search for the presence of known viruses in the human transcriptome [EYS+23, Cao16, Tan13]. Although this type of analysis has yielded results, working with viruses is no easy task. First of all, their genomes evolve rapidly; human papillomavirus (HPV) is an example of a rapidly evolving viral genome. As a result of frequent mutations and recombinations, it presents a high degree of genetic diversity. There are over 100 different strains of papillomavirus, which are classified as high-risk or low-risk according to their association with the development of cancer. A virus associated with a well-known cancer disease may have a large number of strains, and the same applies to new viruses recently associated with a newly discovered cancer. At the moment, not all viral databases and list of viruses targeting humans are complete, adding to the difficulty of identifying cancer-causing viruses. It's a complex task to overcome. Even the simplest solution, such as homology-based methods, is not the most suitable for finding new viruses in cancer sequencing data.

Recently, deep learning methods have been tested to detect bacterial virus sequences in metagenomic sequencing [RSD+20]. In the context of data classification, deep learning models are more and more used in the world as those algorithms succeed in really complex tasks. The domain of bioinformatic is no different, the analysis of genomic sequences has been and is still a difficult problem to solve. This is why, in order to identify the presence of viral genomes in cancer, Elbasir A. et al [EYS+23] decided to try this method.

1.4 DeepDoggo, a deep learning method to detect virus in dog cancer

Here, we develop a method named "DeepDoggo" that uses a deep learning model to distinguish viral reads in dog (Canis lupus familiaris) cancer RNA data. Our project is to replicate the model used in Elbasir et al paper to see what results we can get. As our knowledge in deep learning models is limited, we won't try to improve on the model used.

We chose to focus on the dog model, man's best friend, because access to human data is really difficult. Human cancer RNA databases are often subject to access controls and restrictions. There are several reasons for this. Cancer RNA databases can contain sensitive and personally identifiable informations about patients, including their genomic data. To protect patient privacy and respect ethical considerations, access to these databases is generally controlled. The second reason is data-sharing agreements. The data contained in cancer RNA databases often come from large-scale collaborative research efforts involving several institutions and researchers. Data-sharing agreements and collaborations require specific permissions and restrictions to ensure responsible and appropriate use of the data. Finally, it was not possible for us to access any of the cancer RNA databases, as they all had to comply with legal and regulatory data protection requirements. We therefore overcame this limitation by working on the domestic dog (Canis Lupus Familiaris).

As part of our research and in order to get as close as possible to ViRNAtrap methods, we have selected canine cancers that can be found in humans, including: Large B-cell diffuse lymphoma (DLBCL), prostate cancer, jaw cancer, breast cancer and osteosarcoma. We have also selected two cancers of particular interest in dogs: Hemangiosarcoma and Canine venereal tumor, also known as canine transmissible venereal tumor (CTVT). The Canine venereal tumor is a contagious cancerous tumor that affects dogs. It is transmitted through direct contact during mating or other close interactions between dogs. CTVT originates from malignant transformation of histiocytes, specialized immune cells. It is one of the few known examples of contagious cancers, with tumor cells evading immune recognition [DU13]. Contagious cancers are rare in the wild, the only referenced case being that of facial tumors in the Tasmanian devil, a marsupial (somewhat similar to the dog). This begs the question: is it possible to find viral sequences in samples associated with the Canine venereal tumor?

It is important to mention that the deep learning model we created is not the same as the model used in the paper. As the code for the model creation in TensorFlow has not been found, the model might have slight differences that we will address in the section 3.

2 Pre-processing

2.1 Cancer data collection

Searches were carried out in various databases such as: the european nucleotide archives, Sequence Read Archive (SRA) and Pubmed. For each of the 7 cancer types (lymphoma - diffuse large B-cell lymphoma (DLBCL), prostate cancer, jaw cancer, breast tissue cancer, osteosarcoma, hemangiosarcoma and general transmissible tumor), we selected 6 nucleotide replicates for each cancer. All RNA files were downloaded in Fastq format and checked for quality using FastQC. Only good quality reads were used for the following steps. Next, reads were mapped with Bowtie2 aligned to the Ros-1 canine reference genome (ROS-canfam-01) and to the PhiX phage (NC_001422). This was followed by a read mismatching step using samtools. This step involved several stages, including:

- transformation of bowtie 2 ".sam" files into ".bam" files;
- unmapping of reads;
- sorting of reads.

Reads not mapped to ".bam" format were transformed back to fastq. Finally, the paired-end files were merged and transformed into a fasta file to be used as input for DeepDoggo. Data access, accession numbers and full metadata are available on our github repository.

2.2 Viral database

To train our model, we need to build a database of known viruses that infect dogs. To do this, we consulted "Virus-Host DB", a database that compiles information on the relationships between viruses and certain taxa, using their NCBI taxonomic identifier. For each taxon, the database lists viral reference genomes via Refseq : In our case, we found 74 dog-associated viruses. All viral sequences were compiled in a single file. Among the entries found, various variants were found such as: papillomavirus 21, papillomavirus 22 and 11. All the variants were also included in the file.

2.3 Cutting sequences to 50 bp

Before we feed the data to our deep learning model, we need all of it to be formatted the right way. We made our model so that the genomic sequences should be written line by line but more importantly we specifically need the size of each sequence to be 50 bp long.

To do so we first decided to write a script that would find each genomic sequence in a fasta file and just split the data 50 bp by 50 bp with no overlap at all. If there was a rest, it would just be discarded.

Once we got our three files written (viral, dog and cancer), we started to check the difference in number of genomic sequences between viral and non-viral reads. After evaluation we had over 100 times more non-viral reads than viral ones. For this reason we decided to redo the splitting but this time by adding some overlap. In fact after testing with a window size of 5 and window size of 2 we concluded that we would just take a window size of 1 for our viral reads. Even with that, we still had over 40 times more non-viral reads than viral reads. Because the size of our two dataset are not equal this could have an influence on the training but we decided to let it this way as we couldn't get more viral sequences.

3 Deep Learning model

This part focuses on the conception of the deep learning model, its training and its testing. The model was created and trained using the TensorFlow and Keras libraries in Python.

3.1 Data processing

Before we get into the data processing it is important to note that the total quantity of data far exceed the computer's memory we were using. For this reason all data must be stored as tensors, a special variable type of TensorFlow that does not require to be loaded inside the RAM. This will have an impact on the manipulation we can make as tensor objects are harder to iterate over.

3.1.1 Data extraction

The first part of the deeplearning is to extract the data from the files. For learning, we use two files as input where one file has the viral sequences and the other has the non viral sequences, both files are successions of genomic sequences of 50 bp line by line.

While extracting the files we need to make sure to mix them up as much as we can, it is important for the training to have an evenly spread dataset and it is impossible to completely shuffle our dataset from start to end. To do so we first extract the data of each file and store them in two different dataset, and then we combine them together using a tensorflow function that randomly picks the next sample in each set. As it is possible to indicate the weight (probability) of each set to be choosen for the next sample, we calculated the size of our datasets and its repartition to come with these numbers $viral = 2.6\%$, $nonviral = 97.4\%$. Before doing so we also need to add to each sample the class to which they belong, this can be simply added to each sample before merging.

3.1.2 One-hot encoding

Once the merging has been done, we need to one-hot encode our dataset, meaning transforming the letters into simple vectors where each letter correspond to a different index. The encoding will be done like the following :

- A : [1, 0, 0, 0]
- T : [0, 1, 0, 0]
- C : [0, 0, 1, 0]
- G : [0, 0, 0, 1]

Our final dataset will be of size (50, 4). It was possible to put all of these in the same dimension and end up with a size (200, 1) but without prior knowledge on the paper's solution, we decided on the first one by following the idea of Ren J et al. [RSD+20]. This will change a bit the way our model work but it shouldn't have much impact on the results.

3.1.3 Split

We now need to split the dataset between training and testing. As we won't change the hyper-paremeters, we do not need a third validation test. To do this we need to shuffle our dataset as much as we can first. It is impossible for us to shuffle all the dataset from start to end as it would require to load it all in the RAM, but we can indicate the maximum size of the buffer we use to try to make it the more shuffled possible.

Once the dataset is shuffled we will take the first 80% of it for training and the last 20% for testing. We will do this by indicated the total size of the dataset.

3.2 Model

After some research on the code, it was impossible to find the model creation and training from the authors of the original paper. If the paper described gave the big lines of the preprocessing and model creation, it leaves out a lot of details. It was still possible to find the pre-trained model in the git project.

The model is store as a .hdf5 file, there are two ways that we found to open it: the first was a tool called HDFview working on windows; and the second was simply by using TensorFlow with Python. By loading the model with TensorFlow and using the summary() method on it, we get the following :

Layer (type)	**Output Shape**	**Param #**
main_input (InputLayer)	(None, 48)	0
embedding_18 (Embedding)	(None, 48, 25)	1200
conv1d_18 (Conv1D)	(None, 48, 64)	17664
max_pooling1d_18 (MaxPooling1D)	(None, 48, 64)	0
batch_normalization_18 (BatchNormalization)	(None, 48, 64)	256
dropout_54 (Dropout)	(None, 48, 64)	0
flatten_18 (Flatten)	(None, 3072)	0
FC3 (Dense)	(None, 64)	196672
dropout_56 (Dropout)	(None, 64)	0
output (Dense)	(None, 1)	65

Explanation The model first goes through an embedding layer that will modify the format of the input into a lower-dimensional space. It will then use a convolutional layer that will find specific characteristics of groups of letters in the data, and will use the max pooling layer to extract those that have the biggest values. The batch normalization layer will then normalize the data. Then a dropout layer will be used to modify the links between the neurons in the model and the result will be flatten into a single dimension. Finally they put a layer of neurons and then a last dropout before putting a single output.

It is worth mentioning that all these layers have different output functions, each neurons will get all the data it receives sum it up and put it inside that said function. Not all functions were described but the convolutional layer uses the ReLU function and the last layer uses the Sigmoïd function. That means that the output is not 0 or 1 but a floating number between 0 and 1.

We generally followed the same layer structure for the model but with some difference mainly for convolution and pooling. Those two were thought as a 1-dimensional layer in the original paper but our data is 2-dimensional so we had to change them. We still used only an horizontal translation for convolution to account the one-hot encoding as a single variable. It is of course important to note that the shape of the data is not the same throughout the entire model.

In the end we end up with this model :

Layer (type)	**Output Shape**	**Param #**
conv2d (Conv2D)	(None, 49, 4, 64)	192
max_pooling2d (MaxPooling2D)	(None, 24, 4, 64)	0
batch_normalization_18 (BatchNormalization)	(None, 24, 4, 64)	256
dropout (Dropout)	(None, 24, 4, 64)	0
flatten (Flatten)	(None, 6144)	0
dense (Dense)	(None, 54)	3331830
dropout_1 (Dropout)	(None, 54)	0
dense_1 (Dense)	(None, 1)	55

3.3 Training and testing

We now need to train the model with the data. The first thing we need to do is to split our data into different batches. Instead of training the model for each data separately, the model will calculate the error of multiple data (those that are in the same batch) at the same time and change the network after that. We will choose a batch size of 64 as a rule of thumb, though it is normally best to try multiple value to decide which one works the best. Once this is done we can start fitting the data into our model. We choose to do 150 epochs for the training, this correspond to the number of time our model will go through the entirety of our dataset for the training. Every 5 epochs we saved a checkpoint of the model in case anything would happen and we saved the model afterwards.

Once all of this is done we can start the testing. To do so we pass our testing data through our trained model. We then loop through those data and count the false positive, false negative, true positive and true negative to calculate both accuracy, recall, precision and f1-score, typical metrics used in classification algorithm.

3.4 Model prediction

Once the model has been trained, we can predict our cancerous data to see whether they are viral or not. We're loading back the model in Python and predicting the output of each genomic sequence. Since we don't have a binary value as the output, we consider a threshold of 0.5 for the data. Above it, the data is considered positive, below it, it is considered negative. If a genomic sequence is predicted as non viral we don't do anything. If they are predicted as viral, we one-hot decode them and write them inside an output file that will be analyzed later.

4 Results

4.1 Reconstruction

The model training phase lasted just over 20h00, during which the model reached the 25/150 epoch. This phase was not 100% completed due to hardware limitations, and we then moved on to the prediction phase, which lasted 5h00. We had the possibility of making it last longer, but for reasons of computing power, we stopped it.

Despite the reduced time, we were able to obtain an output model in the form of a FASTA file containing 50 bp nucleotide sequences. In the file, 10473 sequences were predicted as viral. Before moving on to the evaluation of potential viral sequences, we need to assemble the reads into contigs.This was done using SPades [NBA+13]. The assembler was able to reconstruct 20 sequences, ranging from 959 to 35 bp [AGM+90].

The contigs were then tested on the blastn alignment tool, which compares nucleotide sequences to identify similarities in a nucleotide sequence database.

In the following table, we have summarized the blastn results. For each of the 20 entries, we have taken the best result.

Description	**Scientific name**	**percentage identity**	**length**
Canis lupus familiaris isolate 5T1 mitochondrion, complete genome	Canis lupus familiaris	100.00%	959
Canis lupus mitochondrial partial D-loop,isolate AL3032	Canis lupus familiaris	100.00%	742
Canis lupus familiaris isolate L3669_XJ mitochondrion, partial genome	Canis lupus familiaris	100.00%	694
Canis lupus familiaris isolate 875T mitochondrion, complete genome	Canis lupus familiaris	100.00%	599
Canis lupus familiaris isolate 875T mitochondrion, complete genome	Canis lupus familiaris	100.00%	575
Canis lupus familiaris isolate ZERD000004 mitochondrion, partial genome	Canis lupus familiaris	100.00%	463
Canis lupus familiaris isolate ZERD000004 mitochondrion, partial genome	Canis lupus familiaris	100.00%	460
Canis lupus familiaris isolate CAN002 mitochondrion	Canis lupus familiaris	100.00%	409
Canis lupus dingo 28S ribosomal RNA	Canis lupus dingo	100.00%	272
Canis lupus familiaris breed Labrador retriever chromosome 09a	Canis lupus familiaris breed Labrador retriever	100.00%	72

As can be seen in the table, the results show that the predictions made by the model are all mitochondrial sequences of the dog. It is also noticeable in the predictions that there are sequences related to the Labrador Retriever breed. It is highly likely that some cancer samples were collected from this breed. Furthermore, the presence of Canis lupus dingo, a subspecies of the gray wolf native to Australia, is also observed in the table. Domestic dogs and dingoes are closely related phylogenetically, so this prediction is not surprising.

5 Conclusion

5.1 Results

In the end of our work, after the evaluation of our data on NCBI Blastn database, the results didn't show any viral sequences. All the predictions done by our model were related to dog sequences. There are several reasons for our results. Firstly, it is likely that our initial files were free of sequences predicted as viral. Secondly, the low learning rate our model underwent is an important factor to take into account, as it distorted our results. Finally, it is very likely that our model was not optimally parameterized.

Even if we couldn't get good results by trying to follow their method, we still managed to follow the overall path from start to finish.

5.2 Discussion

Our initial idea was to follow Elbasir et al. [EYS+23] paper. It slowly turned out that it was nearly impossible to get cancer data on humans as those data are extremely secured which is why we moved on the dog. On top of that a lot of the paper is really focused on the results they obtained and the analysis of them. The description of the model used and the methods is not complete and the code that can be found on their github does not include the creation and training of their model. It was left for us to guess a lot of the process used and our model is not perfectly similar to the one used.

Because our programs were really time consuming, we didn't had enough time to finish the training of the deep learning model. After doing at least a day of training we only did 25 epochs out of the 150 so we had to stop there. This process would've taken us multiple days to go through and we had not that much time. For this reason we also couldn't evaluate our final model our testing set, but it was still possible for us to predict some data and analyse them. As the result showed no sign of viral sequences, it is highly probable that our model is really bad. Our lack of knowledge and time prevented us from tuning our hyper-parameters and constructing different layers but that could have been a better solution.

Finally the paper itself recognizes the existence of other algorithms that have achieved good results as well sometimes even better such as DeepVirFinder, proving that there are still many ways to implement deep learning into the analysis of genomic sequences.

References

[AGM+90] Stephen F. Altschul, Warren Gish, Webb Miller, Eugene W. Myers, and David J. Lipman. Basic local alignment search tool. *Journal of Molecular Biology*, 215(3):403–410, 1990.

[AN17] Noreen M Ahmed F Atif M Fatima Z Bilal Waqar A Akram N, Imran M. Oncogenic role of tumor viruses in humans. *Viral Immunol*, 30(1)(20-27), 2017.

[And10] S Andrews. Fastqc: A quality control tool for high throughput sequence data. *http://www.bioinformatics.babraham.ac.uk/projects/fastqc/*, 2010.

[Aus20] Gussow A. B. Benler S. Wolf Y. I. Koonin E. V Auslander, N. Seeker: alignment-free identification of bacteriophage genomes by deep learning. *Nucleic Acids Res*, 1(48), 2020.

[Ban04] Araujo A. Yamano Y. Taylor G. P Bangham, C. R. Htlv-1-associated myelopathy/tropical spastic paraparesis. *Nature Reviews Disease Primers*, 1(1), 2004.

[Cao16] S. et al Cao. Divergent viral presentation among human tumors and adjacent normal tissues. *Nature Reviews Disease Primers*, 6(28294), 2016.

[Che06] Yang H. I. Su J. Jen C. L. You S. L. Lu S. N. ... Wang L. Y Chen, C. J. Risk of hepatocellular carcinoma across a biological gradient of serum hepatitis b virus dna level. *Jama*, 295(1)(65-73), 2006.

[Den10] Budczies J. von Minckwitz G. Wienert S. Loibl S. Klauschen F. ... Dietel M Denkert, C. Identification and validation of an anthracycline/cyclophosphamide-based chemotherapy response assay in breast cancer. *BMC Cancer*, 18(1)(72), 2010.

[DU13] Das AK Das U. Canine transmissible venereal tumor: A review. *Vet World*, 6(10)(780-785), 2013.

[ES12] H. B El-Serag. Epidemiology of viral hepatitis and hepatocellular carcinoma. *Gastroenterology*, 142(6)(1264-1273), 2012.

[EYS+23] Abdurrahman Elbasir, Ying Ye, Daniel E. Schäffer, Xue Hao, Jayamanna Wickramasinghe, Konstantinos Tsingas, Paul M. Lieberman, Qi Long, Quaid Morris, Rugang Zhang, Alejandro A. Schäffer, and Noam Auslander. A deep learning approach reveals unexplored landscape of viral expression in cancer. *Nature*, 14(785), 2023.

[Gus18] Rescheneder P. Wachutka L Gusev, F. nsight into the single cell rna-seq protocol from the mouse hippocampus. *EBI European Nucleotide Archive*, 2018.

[Har20] Meshorer E. Fishman A Harikumar, A. Gene expression profiling of developing human primary neurons with ips cells from rett syndrome patients. ncbi gene expression omnibus. *GEO Data Accession GSE135183*, 2020.

[Hua21] Murphy D. J. Martin S. D Huang, S. H. Whole blood rna-sequencing analysis of response to nivolumab in metastatic renal cell carcinoma. *NCBI Gene Expression Omnibus*, 2021.

[JJ17] Dennehy JJ. Evolutionary ecology of virus emergence. *Ann N Y Acad Sci*, 1389(1)(124-146), 2017.

[KK21] Trimarchi T. Soler E Kieffer-Kwon, P. Gene expression data from human primary melanoma. *NCBI Gene Expression Omnibus*, 2021.

[Lan12] Salzberg S. L Langmead, B. Fast gapped-read alignment with bowtie 2. *Nature Methods*, 9(4)(357-359.), 2012.

[Li09] Handsaker B. Wysoker A. Fennell T. Ruan J. Homer N. ... Durbin R Li, H. The sequence alignment/map format and samtools. *Bioinformatics*, 25(16)(2078-2079), 2009.

[MA16] O'Malley MA. The ecological virus. *Stud Hist Philos Biol Biomed Sci*, 9(59-71), 2016.

[NBA+13] Sergey Nurk, Anton Bankevich, Dmitry Antipov, Alexey A Gurevich, Anton Korobeynikov, Alla Lapidus, Andrey Prjibelski, Alexey Pyshkin, Alexander Sirotkin, Yuri Sirotkin, et al. Assembling single-cell genomes and mini-metagenomes from chimeric mda products. *Journal of Computational Biology*, 20(10):714–737, 2013.

[NS23] Parveen S Abbass K Song H Achim MV Naseer S, Khalid S. Covid-19 outbreak: Impact on global economy. *Front Public Health.*, 30(10), 2023.

[Qui10] Hall I. M Quinlan, A. R. Bedtools: a flexible suite of utilities for comparing genomic features. *Bioinformatics*, 26(6)(841-842), 2010.

[Ren17] Ahlgren N. A. Lu Y. Y. Fuhrman J. A. Sun F Ren, J. Virfinder: a novel k-mer based tool for identifying viral sequences from assembled metagenomic data. *Microbiome*, 5(69), 2017.

[RM16] Roberts G. W. Naveiras O Rodriguez-Meira, A. Single-cell rna sequencing reveals a pre-erythrocytic erythro-myeloid progenitor lineage in the mouse embryo. *EBI European Nucleotide Archive*, 2016.

[RM19] Buck G. Clark S. A. Povinelli B. J. Alcolea V. Louka E. ... Mead A. J Rodriguez-Meira, A. Unravelling intratumoral heterogeneity through high-sensitivity single-cell mutational analysis and parallel rna sequencing. *Leukemia*, 33(4)(879-894), 2019.

[RSD+20] Jie Ren, Kai Song, Chao Deng, Nathan A. Ahlgren, Jed A. Fuhrman, Yi Li, Xiaohui Xie, Ryan Poplin, and Fengzhu Sun. Identifying viruses from metagenomic data using deep learning. *Quantitative biology*, 8(1), 2020.

[Sal13] N. F Salyakina, D. Tsinoremas. Viral expression associated with gastrointestinal adenocarcinomas in tcga high-throughput sequencing data. *Hum. Genomics*, 7(23), 2013.

[Sie17] Martens J. W. M. Foekens J. A Sieuwerts, A. M. Genomic alterations in breast cancer metastases. *Breast Cancer Research*, 19(1)(81), 2017.

[Tan13] Alaei-Mahabadi B. Samuelsson T. Lindh M. Larsson E Tang, K. W. The landscape of viral expression and host gene fusion and adaptation in human cancer. *Nat. Commun*, 4(2513), 2013.

[You04] Rickinson A. B Young, L. S. Epstein-barr virus: 40 years on. nature reviews cancer. *Gastroenterology*, 4(10)(757-768), 2004.

[zH09] H. zur Hausen. Papillomaviruses and cancer: from basic studies to clinical application. *Nature Reviews Cancer*, 9(3)(235-246), 2009.

[Zha17] Sieuwerts A. M. McGreevy M. Casey G. Cufer T. Paradiso A. ... Wang Y Zhang, Y. The 76-gene signature defines high-risk patients that benefit from adjuvant tamoxifen therapy. *Breast Cancer Research and Treatment*, 166(2)(523-532), 2017.