🧬Enhanced detection of Monkeypox Virus infection using an integrated approach of Deep Learning and Gene Expression profiles

Project Summary Report

Author: Chaima BenMohamed

Contact Information: chaima.benmohamed@insat.ucar.tn

Report Date: August, 2024

Summary

In this project, 18 samples from colon organoids infected with different clades of the monkeypox virus (MPXV) were analyzed to uncover the effects of the virus on gene expression. Differential expression analysis identified several significantly up- or down-regulated genes compared to control samples, with rigorous filtering based on p-values, adjusted p-values, and log fold change (logFC) thresholds. This was followed by gene set enrichment analysis using Enrichr and the WEB-based Gene Set Analysis Toolkit, which revealed key pathways and ontologies affected by the virus.

To further refine the results, hub genes were identified through network analysis using the KEGG database and Cytoscape's Cytohubba algorithms, with 8 out of 11 algorithms validating the hub genes. These insights were leveraged alongside a neural network trained on image data of monkeypox lesions. The image classification model, developed using a dataset containing 228 images of monkeypox and other similar infections, each resized to 224x224 pixels, achieved an accuracy of 0.8768 after data augmentation and architectural improvements.

For detailed information, please read the following report and the attached supplementary data.

Contents

List of Figures

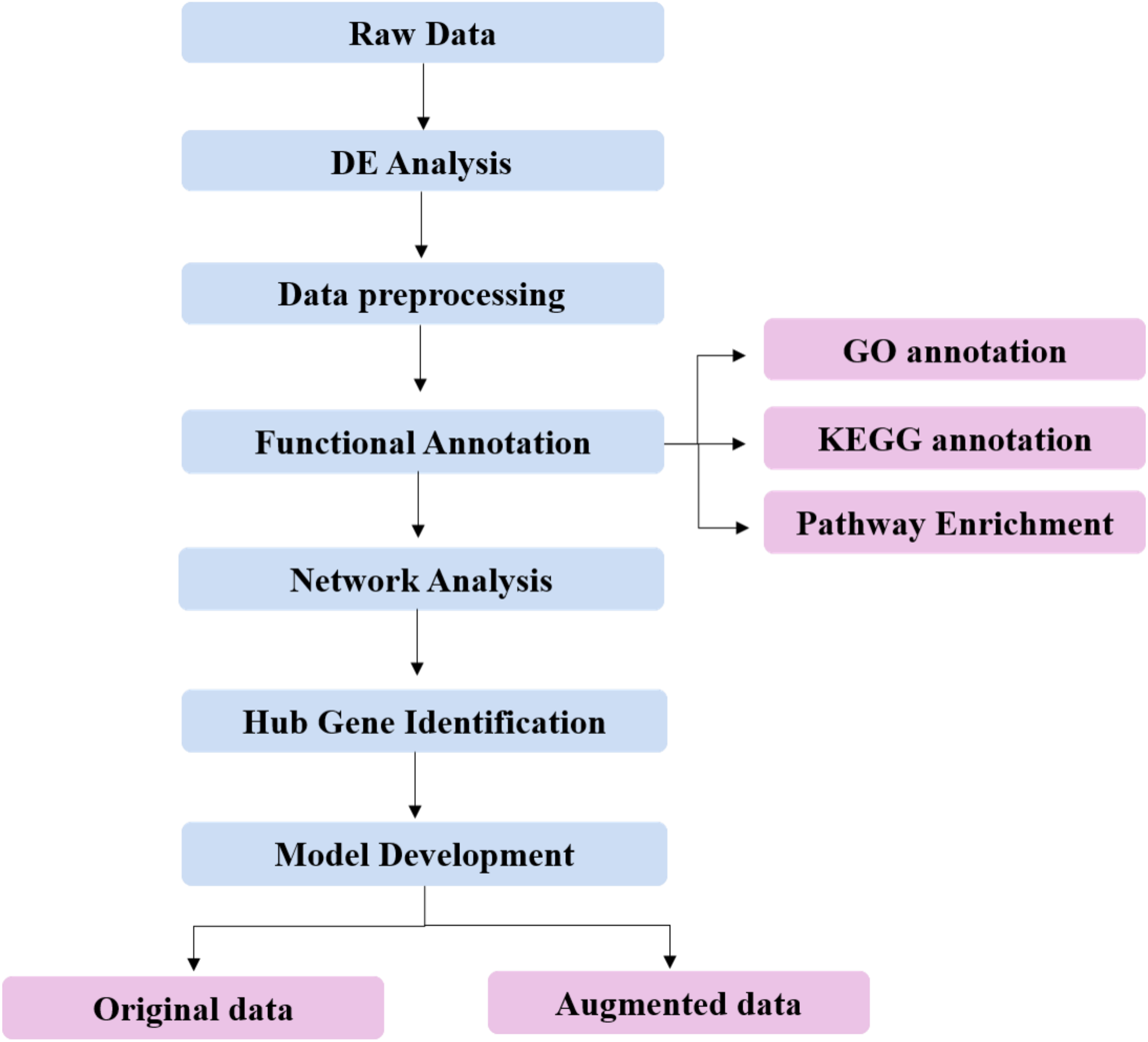
List of tables

List of acronyms

- **MPXV:** Monkeypox Virus
- **GEO:** Gene Expression Omnibus
- **GO:** Gene Ontology
- **KEGG:** Kyoto Encyclopedia of Genes and Genomes
- **DESeq2:** Differential Expression Analysis for Sequence Count Data (R package)
- **GSEA:** Gene Set Enrichment Analysis
- **ER:** Endoplasmic Reticulum
- **CNN:** Convolutional Neural Network
- **JPEG:** Joint Photographic Experts Group
- **ReLU:** Rectified Linear Unit
- **RMSprop:** Root Mean Square Propagation (optimizer)
- **BP:** Biological Process
- **CC:** Cellular Component
- **MF:** Molecular Function
- **MCC:** Maximal Clique Centrality
- **MNC:** Maximum Neighborhood Component
- **EPC:** Edge Percolated Component
- **DMNC:** Degree-based Maximum Neighborhood Component

1- Analysis workflow

The Bioinformatics Analysis workflow is below :



DE: Differential Expression

2- Bioinformatics Analysis

2.1. Raw data overview

The raw data for this project was obtained from the NCBI Gene Expression Omnibus (GEO) database.

The dataset includes both control and infected samples, specifically focusing on colon organoids infected with various clades of the monkeypox virus (MPXV).

Colon organoids serve as a model to study the effects of the monkeypox virus on human intestinal tissues. These 3D cultures mimic the structure and function of the human colon, allowing to observe how the virus interact with the gastrointestinal tract.

Sample Grouping:

- **Control:** 9 samples from uninfected colon organoids.
- **MPXV Clade IIa:** 3 samples.
- **MPXV Clade IIb:** 4 samples.
- **MPXV Clade I:** 3 samples.

2.2 Differential Expression Analysis

To identify genes differentially expressed due to monkeypox infection, I utilized the DESeq2 package in RStudio. I specifically chose DESeq2 for its robustness in handling small sample sizes. This package normalizes count data, estimates dispersion, and applies statistical tests to identify significant changes in gene expression.

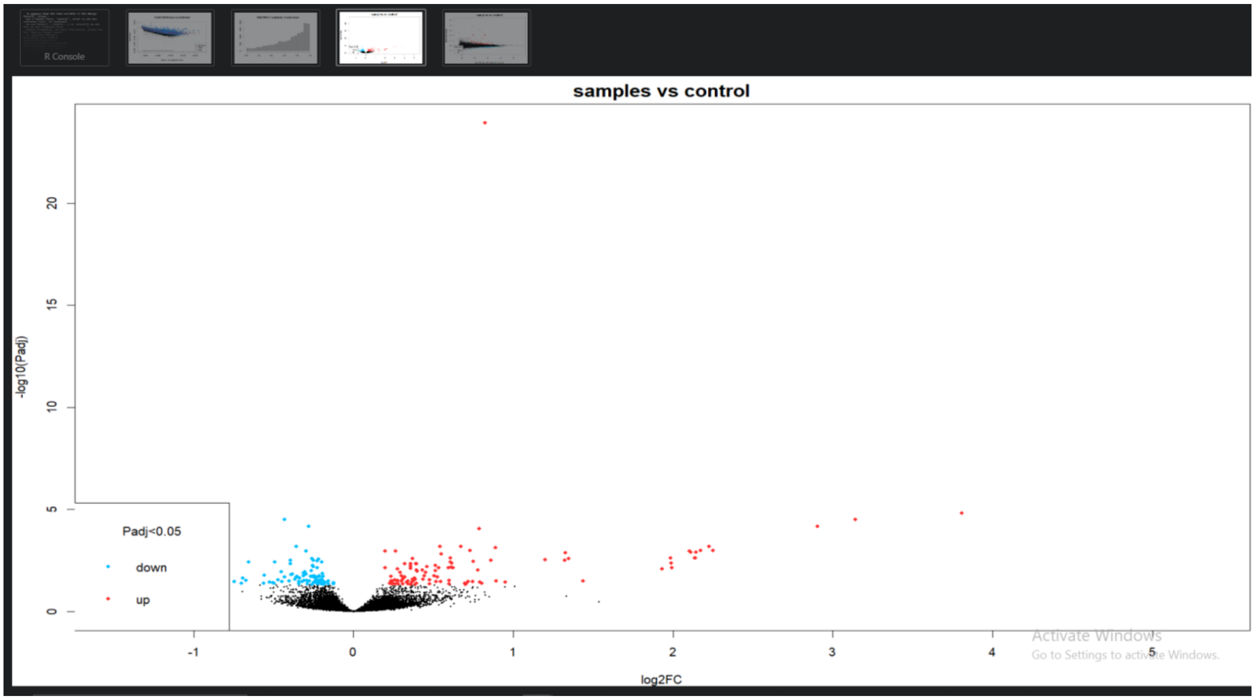


Figure 1: Volcano plot: the distribution of differentially expressed genes

- ✓ Significant genes are highlighted, with red representing up-regulated and blue representing down-regulated genes.

Interpretation: The volcano plot highlights the genes that show statistically significant changes in expression levels due to MPXV infection, providing a visual overview of the extent and direction of differential expression.

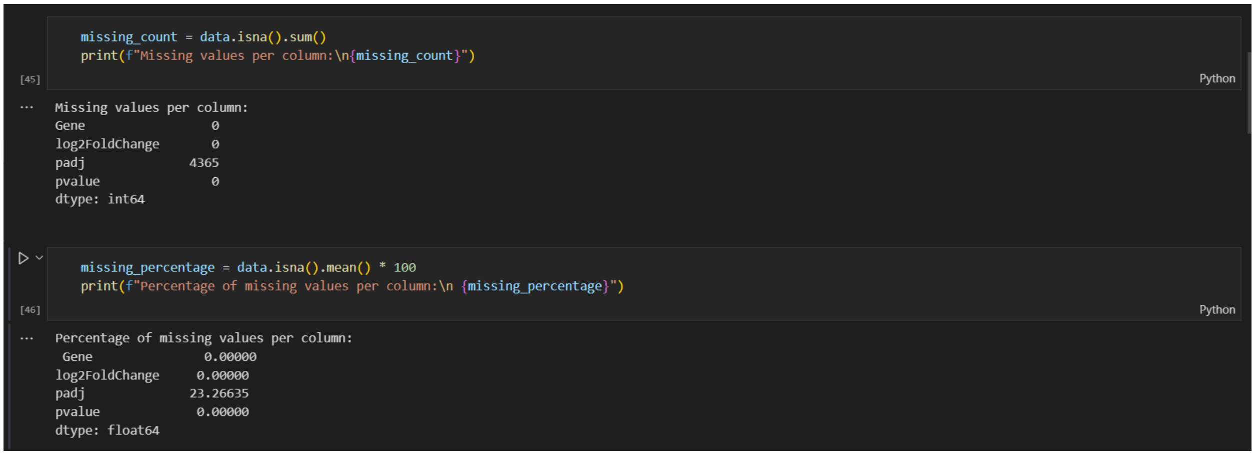
2.3 Gene Expression Data

2.2.1 Data preprocessing

This step involves cleaning and preparing data for analysis by handling missing values and transforming them into a suitable format. It ensures that the data is accurate, consistent, and ready for subsequent processing.

Handling Missing Values:

- Inspection**:** Initially, I examined the dataset for missing values.



```
[45]
missing_count = data.isna().sum()
print(f"Missing values per column:\n{missing_count}")
```

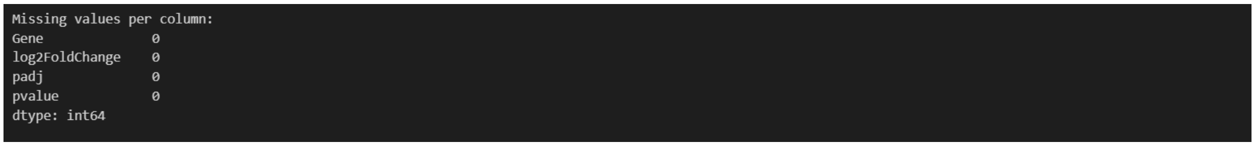
```
Missing values per column:
Gene                0
log2FoldChange      0
padj             4365
pvalue              0
dtype: int64
```

```
[46]
missing_percentage = data.isna().mean() * 100
print(f"Percentage of missing values per column:\n {missing_percentage}")
```

```
Percentage of missing values per column:
 Gene               0.00000
log2FoldChange     0.00000
padj              23.26635
pvalue             0.00000
dtype: float64
```

Figure 2: Handling missing values

- **Filtering:** Missing values were removed.

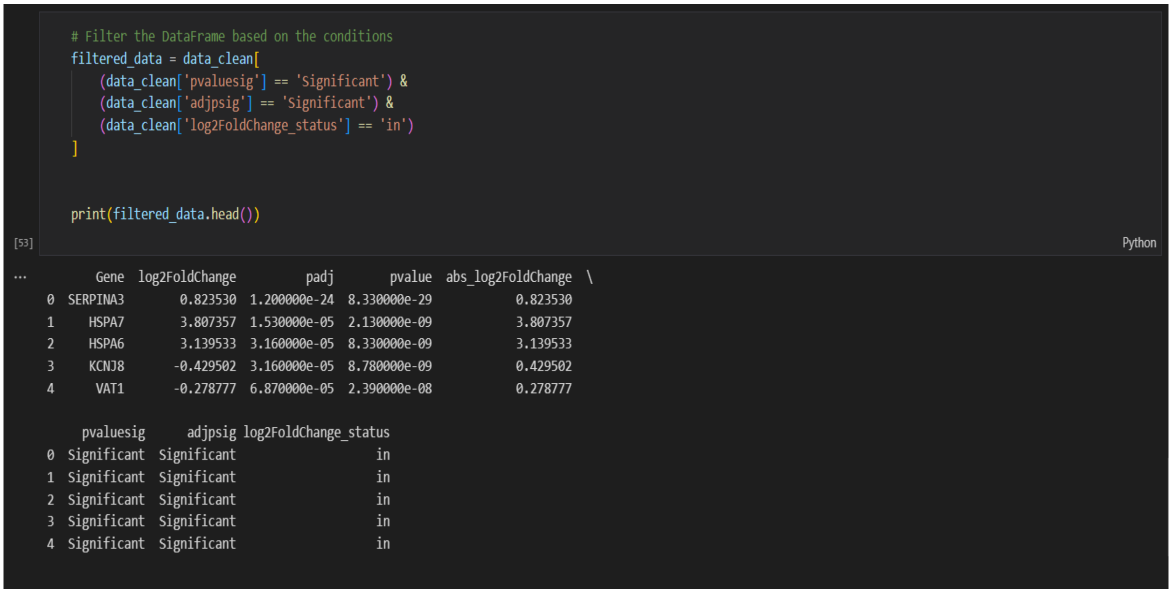


```
Missing values per column:
Gene              0
log2FoldChange    0
padj              0
pvalue            0
dtype: int64
```

Figure 3: Missing values removed

Filter criteria applied: (see figure 4)

- **P-value Threshold:** The genes with a Pvalue< 0.05 were filtered.
- **Adjusted P-value Threshold:** The genes with an adjusted Pvalue <0.05 were removed.
- **Log Fold Change Threshold:** Only genes with absolute logFC > 0.1 were retained to ensure relevance.



```python
# Filter the DataFrame based on the conditions
filtered_data = data_clean[
    (data_clean['pvaluesig'] == 'Significant') &
    (data_clean['adjpsig'] == 'Significant') &
    (data_clean['log2FoldChange_status'] == 'in')
]


print(filtered_data.head())
```

[53] Python

```
      Gene  log2FoldChange          padj        pvalue  abs_log2FoldChange  \
0  SERPINA3        0.823530  1.200000e-24  8.330000e-29            0.823530
1     HSPA7        3.807357  1.530000e-05  2.130000e-09            3.807357
2     HSPA6        3.139533  3.160000e-05  8.330000e-09            3.139533
3     KCNJ8       -0.429502  3.160000e-05  8.780000e-09            0.429502
4      VAT1       -0.278777  6.870000e-05  2.390000e-08            0.278777

     pvaluesig      adjpsig log2FoldChange_status
0  Significant  Significant                    in
1  Significant  Significant                    in
2  Significant  Significant                    in
3  Significant  Significant                    in
4  Significant  Significant                    in
```

Figure 4: Filter criteria

Data Dimensions:

- **Total Genes Analyzed:** 18762 genes were identified for this project.
- **Number of significant genes:** 203 were found to have a significant differential expression

2.2.2 Data exploration

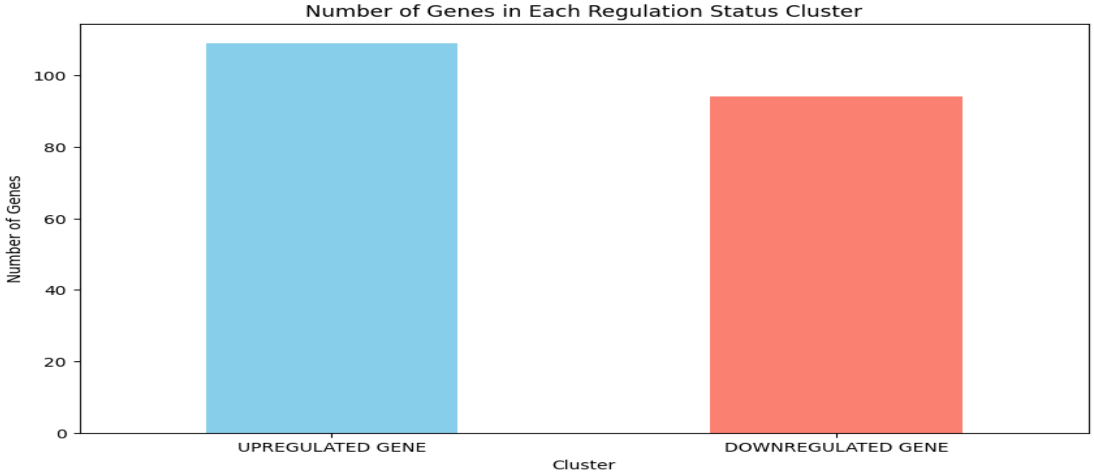


Figure 5: Number of upregulated genes vs downregulated genes

Table 1: Summary of differentially expressed genes

Cluster name	Up-regulated	Down-regulated
Number of genes	109	94

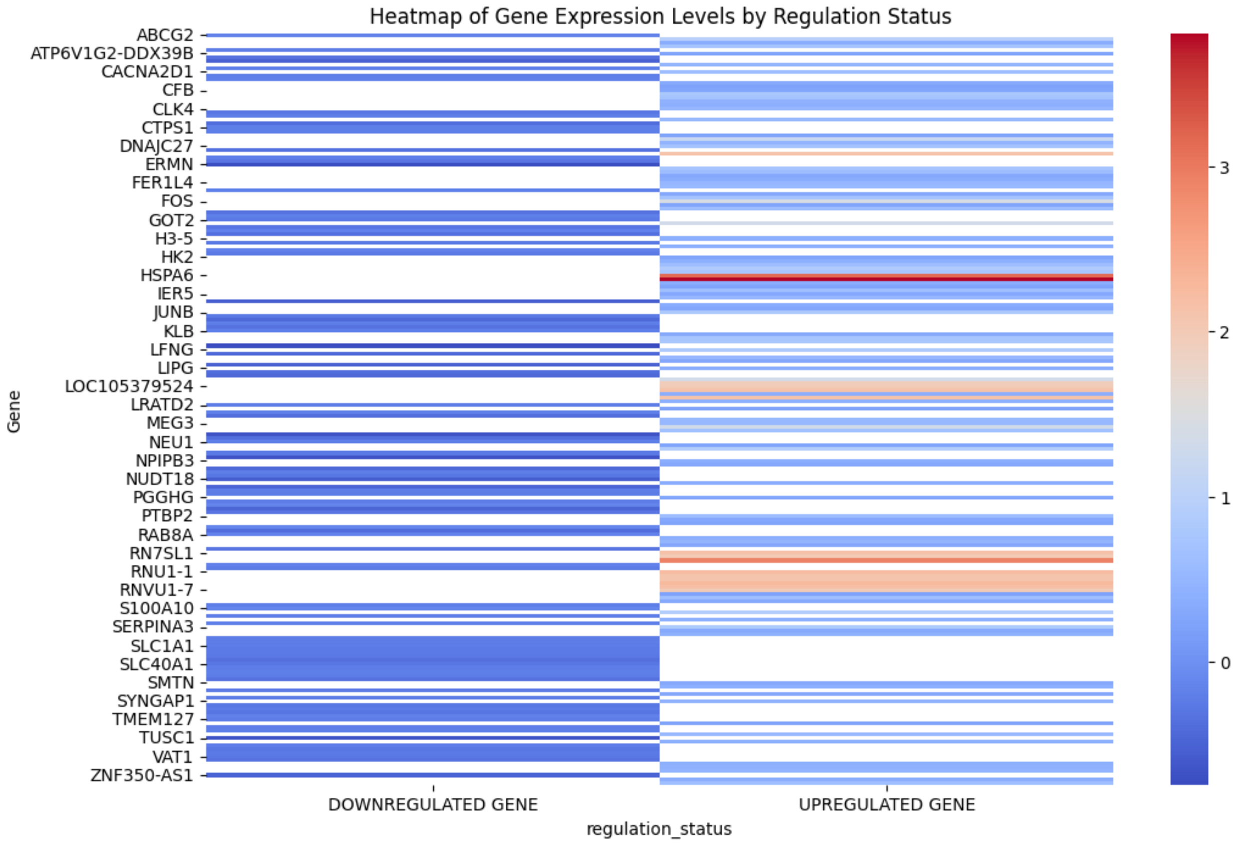


Figure 6: Heatmap of the expression levels of differentially expressed genes across all samples

- ✓ The color gradient represents the level of gene expression, with darker colors indicating higher expression levels.

2.4 Gene set enrichment and annotation

To understand the functions and biological processes associated with the identified genes, I performed functional annotation using Enrichr and the WEB-based Gene Set Analysis Toolkit, focusing on KEGG pathways and Gene Ontology (GO) biological processes.

2.4.1 GO annotation

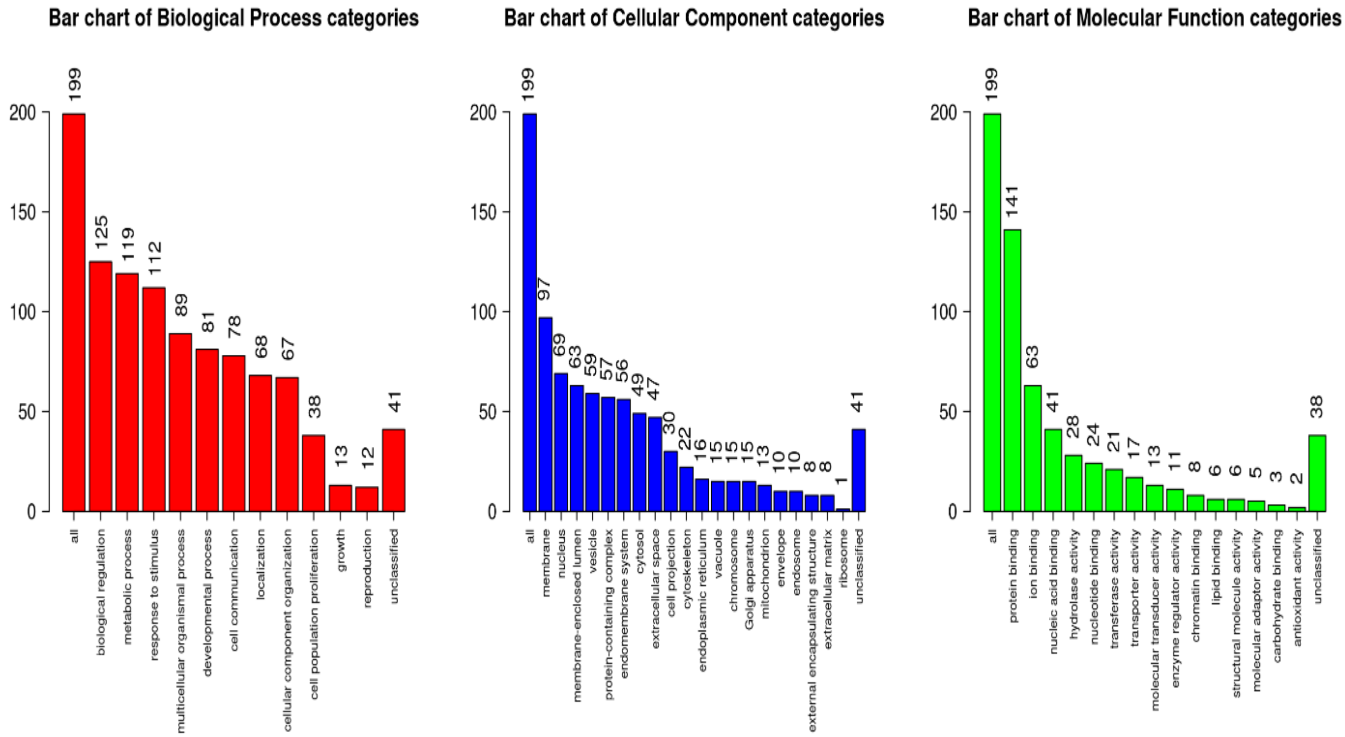


Figure 7: GO annotation of the differentially expressed genes

The bar charts represent the results of gene set enrichment analysis (GSEA) for the monkeypox infection, categorized into three main Gene Ontology (GO) terms: Biological Process (BP), Cellular Component (CC), and Molecular Function (MF).

1- **Biological Process (BP) Categories**:

- **Biological regulation** (199 terms): This category is pivotal in maintaining cellular and systemic homeostasis, biological regulation is what allows an organism to handle the effects of a perturbation, modulating its constitutive dynamics in response to particular changes in internal and external conditions. The monkeypox virus can hijack regulatory pathways to evade the immune response, promoting viral replication and spread.

- **Metabolic processes** (125 terms): The virus often alters host metabolic processes to favor viral replication. For example, changes in lipid metabolism can assist in viral envelope formation, crucial for monkeypox virus assembly and egress.

2- **Cellular Component (CC) Categories**:

- **Membranes** (97 terms): The high number of terms related to membranes suggests that the virus interacts with or disrupts cellular membranes, including endosomal or plasma membranes, which could facilitate viral entry, replication, or release.
- **Intracellular and extracellular organelles**: These include mitochondria, endoplasmic reticulum, and lysosomes. Monkeypox virus exploits these organelles for replication (e.g., by altering ER functions to enhance protein synthesis for viral components).

3- **Molecular Function (MF) Categories**:

- **Protein binding** (141 terms): Protein-protein interactions are crucial in viral pathogenesis. Monkeypox virus could bind to host proteins to manipulate cellular machinery, evade immune responses, or facilitate its replication cycle.
- **Hydrolase and transferase activities** (63 and 41 terms, respectively): These enzyme activities are often co-opted by viruses to modify host cell metabolism, promoting viral replication and assembly.

➢ The monkeypox infection in colon organoids affects several key cellular processes, primarily related to the stress response and the immune modulation such as Th17 cell differentiation.

2.4.2 KEGG Annotation

The Kyoto Encyclopedia of Genes and Genomes (KEGG) facilitates the exploration of gene functions within the broader context of cellular and molecular processes by mapping genes to known metabolic and signaling pathways.

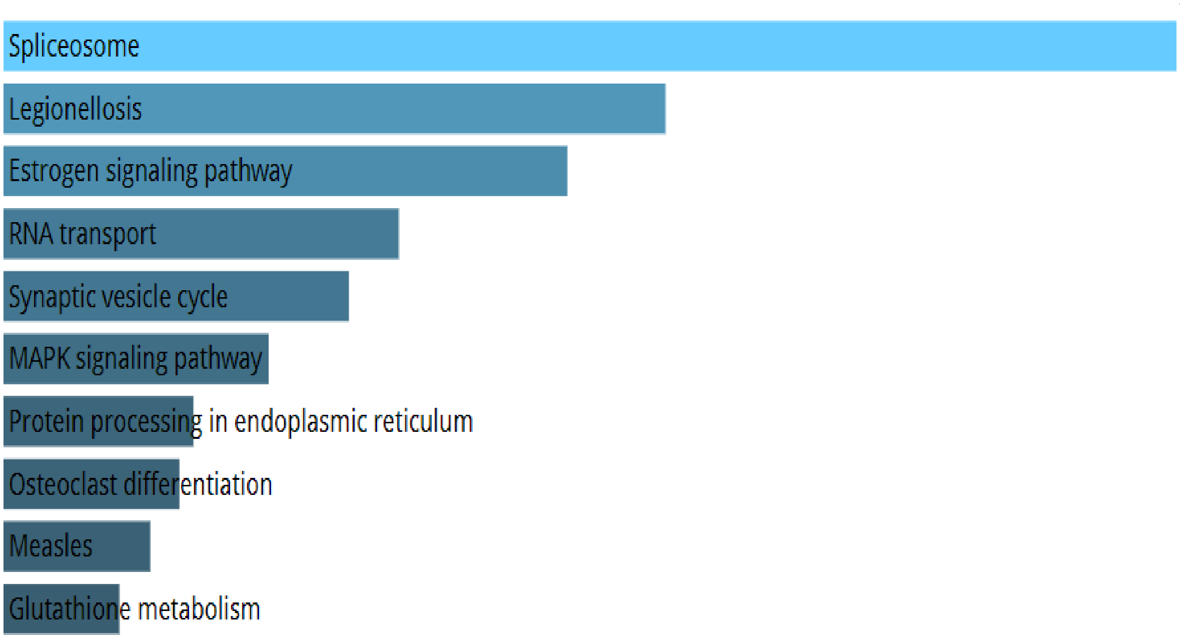
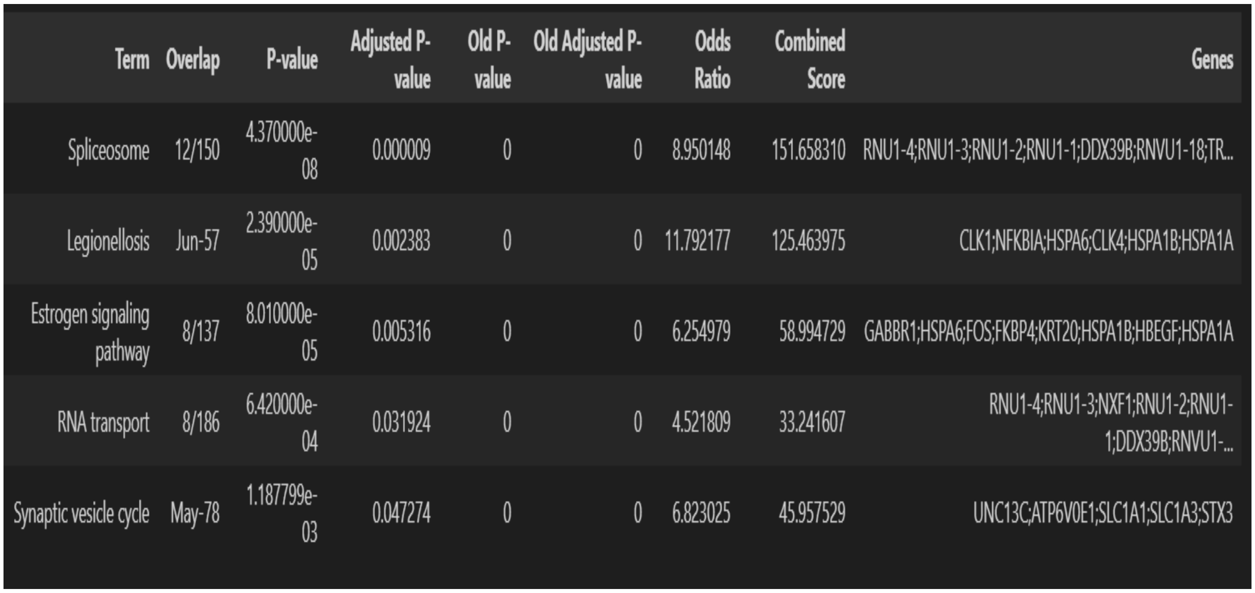


Figure 8: Key pathways implicated



Term	Overlap	P-value	Adjusted P-value	Old P-value	Old Adjusted P-value	Odds Ratio	Combined Score	Genes
Spliceosome	12/150	4.370000e-08	0.000009	0	0	8.950148	151.658310	RNU1-4;RNU1-3;RNU1-2;RNU1-1;DDX39B;RNVU1-18;TR...
Legionellosis	Jun-57	2.390000e-05	0.002383	0	0	11.792177	125.463975	CLK1;NFKBIA;HSPA6;CLK4;HSPA1B;HSPA1A
Estrogen signaling pathway	8/137	8.010000e-05	0.005316	0	0	6.254979	58.994729	GABBR1;HSPA6;FOS;FKBP4;KRT20;HSPA1B;HBEGF;HSPA1A
RNA transport	8/186	6.420000e-04	0.031924	0	0	4.521809	33.241607	RNU1-4;RNU1-3;NXF1;RNU1-2;RNU1-1;DDX39B;RNVU1-...
Synaptic vesicle cycle	May-78	1.187799e-03	0.047274	0	0	6.823025	45.957529	UNC13C;ATP6V0E1;SLC1A1;SLC1A3;STX3

Figure 9: Summary of KEGG pathway annotation

Spliceosome:

The spliceosome is responsible for the splicing of pre-mRNA, a critical step in gene expression. Viral interference with splicing could lead to the production of aberrant host proteins, aiding in immune evasion or modifying the cellular environment to favor viral replication.

Legionellosis:

This pathway highlights cellular processes that could be co-opted by viruses, such as vesicle trafficking and immune evasion mechanisms. Monkeypox might use similar pathways to evade detection and manipulate the host's intracellular environment.

Estrogen Signaling Pathway:

Estrogen signaling can modulate immune responses and cellular proliferation. Viruses may alter this pathway to suppress immune responses or influence cellular proliferation, which could enhance viral replication and spread.

RNA Transport:

Efficient RNA transport is crucial for viral replication, especially for RNA viruses. Even though monkeypox is a DNA virus, alterations in host RNA transport mechanisms could impact the processing and translation of viral mRNAs.

MAPK Signaling Pathway:

The MAPK pathway is involved in regulating cell growth, apoptosis, and immune responses. Viruses often manipulate this pathway to prevent apoptosis of infected cells, ensuring prolonged survival and viral replication.

Synaptic Vesicle Cycle:

Though primarily related to neuronal function, alterations in vesicle trafficking can affect viral entry, replication, and egress. Viruses might exploit this pathway to facilitate their movement within cells or between cells.

Protein Processing in the Endoplasmic Reticulum (ER):

The ER is a critical site for protein folding and modification. Viral infection can induce ER stress, leading to the unfolded protein response (UPR), which viruses often manipulate to promote their replication.

Osteoclast Differentiation:

While not directly related to viral infection, the involvement of osteoclast differentiation could suggest viral manipulation of bone remodeling processes or immune cells associated with bone tissue.

Glutathione Metabolism:

Glutathione is essential for managing oxidative stress. Viral infections often induce oxidative stress, and manipulating glutathione metabolism can either protect the virus from oxidative damage or enhance the stress to kill immune cells.

- The virus targets essential regulatory, metabolic, and signaling pathways, using or disrupting them to create a favorable environment for its lifecycle. The specific manipulation of processes like splicing, protein processing, and signal transduction is particularly crucial for the virus to maintain its replication within the host while **avoiding detection by the immune system**.

2.5 Network analysis and Hub Gene identification

To enhance the clarity of the KEGG network visualization, I applied the "hide disconnected nodes" filter to focus on the most relevant interactions. The resulting network highlighted the most significant genes and their interactions within the KEGG pathways (figure 10).

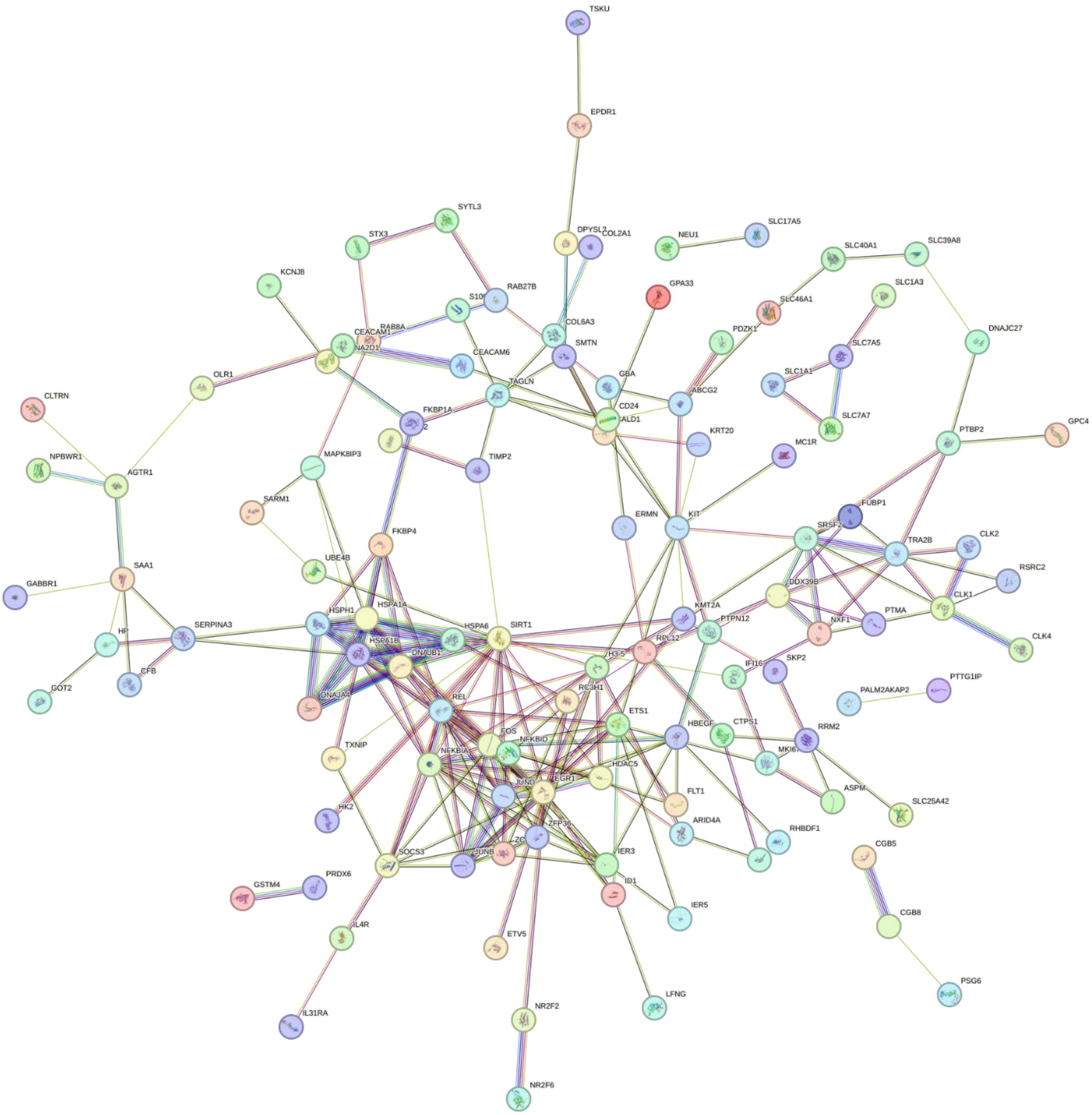


Figure 10: KEGG network visualization after filtering out disconnected nodes

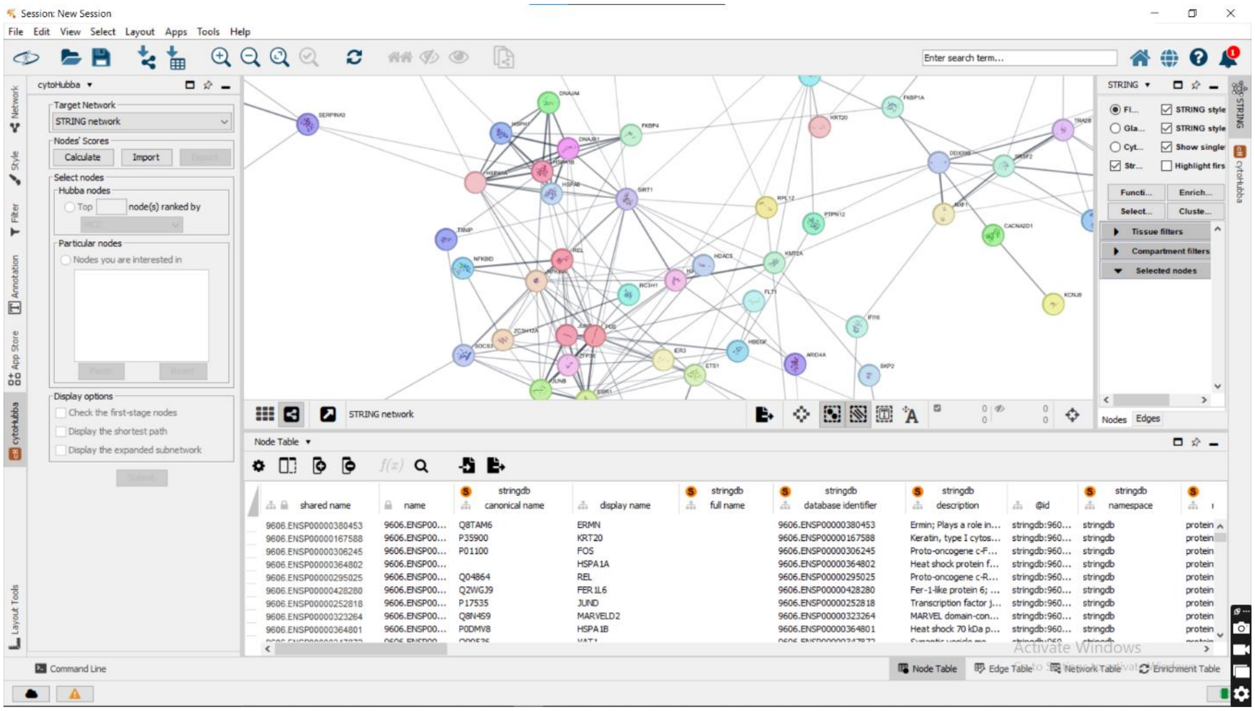
Interpretation: This network visualization emphasizes the core genes that are central to the biological processes triggered by MPXV infection, providing a clearer understanding of gene interactions.

Following the functional annotation and KEGG network visualization, I identified key regulatory genes, known as hub genes, using Cytoscape's CytoHubba plugin. I employed eight algorithms to cross-validate the importance of these genes in the network

What is a hub gene?

Hub gene: A gene that is highly connected to other genes in a network, and is thought to be important for the regulation of the network as a whole. (Current Opinion in Plant Biology, 2023)

With a refined network, I first imported the KEGG network data into Cytoscape, a powerful network visualization and analysis tool (figure.).



shared name	name	stringdb canonical name	display name	stringdb full name	stringdb database identifier	stringdb description	@id	stringdb namespace	
9606.ENSP00000380453	9606.ENSP00...	Q8TAM6	ERMN		9606.ENSP00000380453	Ermin; Plays a role in...	stringdb:960...	stringdb	protein
9606.ENSP00000167588	9606.ENSP00...	P35900	KRT20		9606.ENSP00000167588	Keratin, type I cytos...	stringdb:960...	stringdb	protein
9606.ENSP00000306245	9606.ENSP00...	P01100	FOS		9606.ENSP00000306245	Proto-oncogene c-F...	stringdb:960...	stringdb	protein
9606.ENSP00000364802	9606.ENSP00...		HSPA1A		9606.ENSP00000364802	Heat shock protein f...	stringdb:960...	stringdb	protein
9606.ENSP00000295025	9606.ENSP00...	Q04864	REL		9606.ENSP00000295025	Proto-oncogene c-R...	stringdb:960...	stringdb	protein
9606.ENSP00000428280	9606.ENSP00...	Q2WGJ9	FER1L6		9606.ENSP00000428280	Fer-1-like protein 6; ...	stringdb:960...	stringdb	protein
9606.ENSP00000252818	9606.ENSP00...	P17535	JUND		9606.ENSP00000252818	Transcription factor j...	stringdb:960...	stringdb	protein
9606.ENSP00000323264	9606.ENSP00...	Q8N4S9	MARVELD2		9606.ENSP00000323264	MARVEL domain-con...	stringdb:960...	stringdb	protein
9606.ENSP00000364801	9606.ENSP00...	P0DMV8	HSPA1B		9606.ENSP00000364801	Heat shock 70 kDa p...	stringdb:960...	stringdb	protein

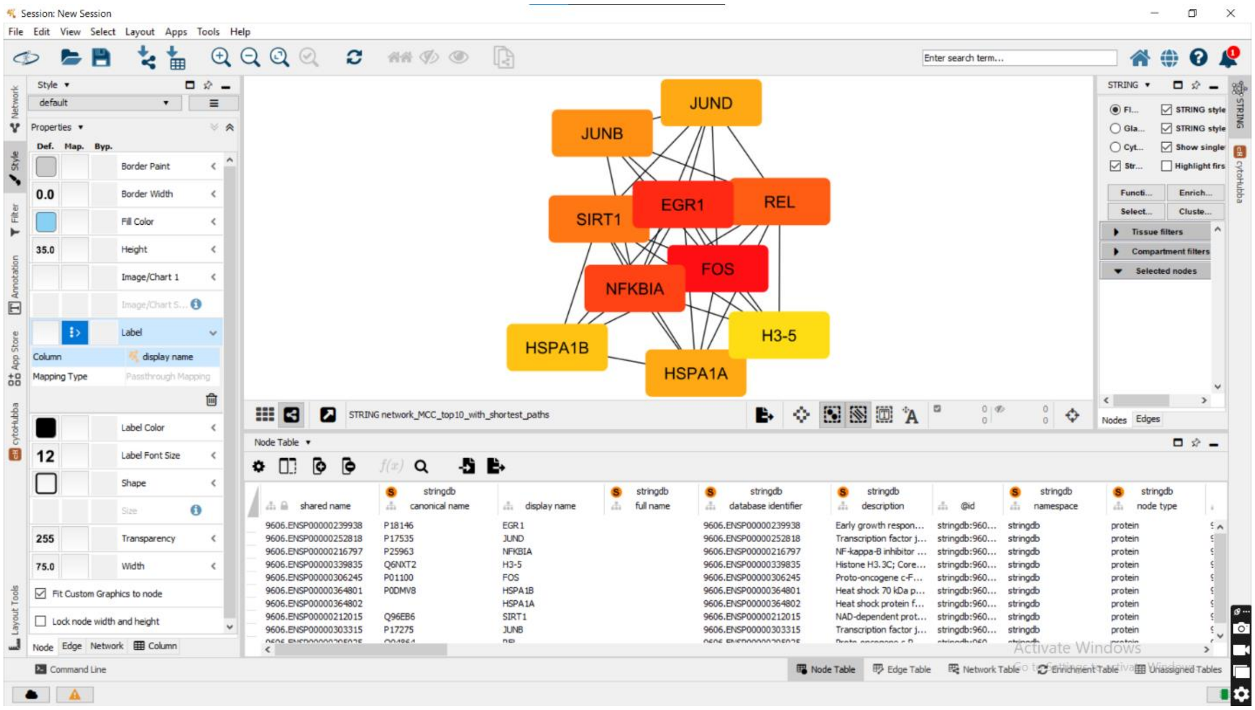
Figure 11 Network visualization of the monkeypox infection-related genes using Cytoscape

Then, I employed Cytoscape’s CytoHubba plugin to identify hub genes. I used the results from eight out of the eleven algorithms provided by CytoHubba:

4- **MCC (Maximal Clique Centrality)**
5- **MNC (Maximum Neighborhood Component)**
6- **EPC (Edge Percolated Component)**
7- **EcCentricity (Eccentricity Centrality)**
8- **DMNC (Degree-based Maximum Neighborhood Component)**
9- **Degree (Node Degree)**
10- **Closeness (Closeness Centrality)**
11- **Bottleneck (Bottleneck Centrality)**

Each algorithm assessed gene importance from different perspectives, providing a comprehensive evaluation of gene centrality.

I analyzed the results from these algorithms to identify genes consistently marked as central across multiple metrics. This approach ensured that the selected hub genes were critical to the network’s structure and function.



Style: default

Def.	Map.	Byp.	Properties
			Border Paint
0.0			Border Width
			Fill Color
35.0			Height
			Image/Chart 1
			Image/Chart S...
			Label

Column: display name
Mapping Type: Passthrough Mapping

Def.	Map.	Byp.	Properties
			Label Color
12			Label Font Size
			Shape
			Size
255			Transparency
75.0			Width

☑ Fit Custom Graphics to node
□ Lock node width and height

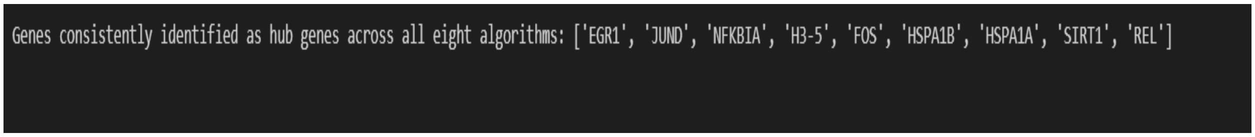
STRING: ● Fl... ○ Gla... ○ Cyt... ☑ Str... ☑ STRING style ☑ STRING style ☑ Show single □ Highlight firs

STRING network_MCC_top10_with_shortest_paths

shared name	stringdb canonical name	display name	stringdb full name	stringdb database identifier	stringdb description	@id	stringdb namespace	stringdb node type
9606.ENSP00000239938	P18146	EGR1		9606.ENSP00000239938	Early growth respon...	stringdb:960...	stringdb	protein
9606.ENSP00000252818	P17535	JUND		9606.ENSP00000252818	Transcription factor j...	stringdb:960...	stringdb	protein
9606.ENSP00000216797	P25963	NFKBIA		9606.ENSP00000216797	NF-kappa-B inhibitor ...	stringdb:960...	stringdb	protein
9606.ENSP00000339835	Q6NXT2	H3-5		9606.ENSP00000339835	Histone H3.3C; Core...	stringdb:960...	stringdb	protein
9606.ENSP00000306245	P01100	FOS		9606.ENSP00000306245	Proto-oncogene c-F...	stringdb:960...	stringdb	protein
9606.ENSP00000364801	P0DMV8	HSPA1B		9606.ENSP00000364801	Heat shock 70 kDa p...	stringdb:960...	stringdb	protein
9606.ENSP00000364802		HSPA1A		9606.ENSP00000364802	Heat shock protein f...	stringdb:960...	stringdb	protein
9606.ENSP00000212015	Q96EB6	SIRT1		9606.ENSP00000212015	NAD-dependent prot...	stringdb:960...	stringdb	protein
9606.ENSP00000303315	P17275	JUNB		9606.ENSP00000303315	Transcription factor j...	stringdb:960...	stringdb	protein

Figure 12: Hub genes based on their centrality scores across different CytoHubba algorithms

To enhance the reliability of the hub gene identification, I cross-referenced the hub genes identified by all eight algorithms. This validation process confirmed the significance of these genes in the context of the monkeypox infection network (figure 13).



```
Genes consistently identified as hub genes across all eight algorithms: ['EGR1', 'JUND', 'NFKBIA', 'H3-5', 'FOS', 'HSPA1B', 'HSPA1A', 'SIRT1', 'REL']
```

Figure 13: List of hub genes

- ⇨ The identified hub genes represent central nodes in the KEGG network. These genes are key regulatory elements involved in the disease.
- ⇨ The identified hub genes will be integrated into the deep learning models developed for monkeypox classification.

- ➢ The virus influences different biological processes and pathways, which manifests in both the internal (colon) and external (skin) phenotypes of the infected patients.

The skin lesions characteristic of monkeypox infections result from viral replication and immune responses in the skin. The pathways and GO terms identified provide insights into how these skin phenotypes develop:

1. **Spliceosome and RNA Transport**: Efficient splicing and RNA transport are essential for producing viral proteins that can lead to cell lysis and the formation of lesions. Disruption in these pathways by the virus could result in abnormal protein production, contributing to the characteristic pox lesions.
2. **Estrogen Signaling Pathway**: Estrogen can influence skin healing and immune responses. The virus exploits this pathway to modulate local immune responses in the skin, potentially delaying the resolution of lesions or influencing the severity of the infection.
3. **MAPK Signaling Pathway**: This pathway is involved in inflammatory responses and cell proliferation. In skin cells, activation of MAPK by the virus could lead to increased inflammation and proliferation of skin cells, contributing to the formation of the characteristic pustules and vesicles seen in monkeypox infections.

➢ **Connection Between Colon and Skin Phenotypes**

Many of the identified pathways, such as **MAPK signaling** and **protein processing**, are relevant in both the skin and the gut. This indicates that the virus uses similar mechanisms to induce pathologies in different tissues, leading to the diverse phenotypes observed in infected patients.

Immune Response: Both the gut and skin are critical components of the body's immune defense. The virus’s ability to modulate immune responses in these tissues could explain the persistence of symptoms in both areas, including the chronic nature of the skin lesions and possible gastrointestinal symptoms

3- Model Development

3.1 Model for Original Data

3.1.1 Dataset Overview

Initially, the dataset contained 228 images: 102 with monkeypox infections and 126 with other infections resembling monkeypox.

The data was split into:

- Training Set: 128 images
- Test Set: 57 images
- Validation Set: 43 images

The pictures are medium-resolution color JPEGs. Figure 14 shows some examples.

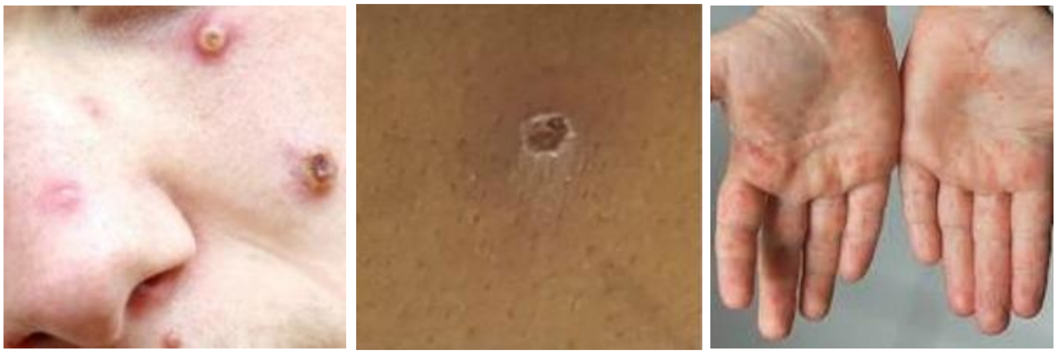
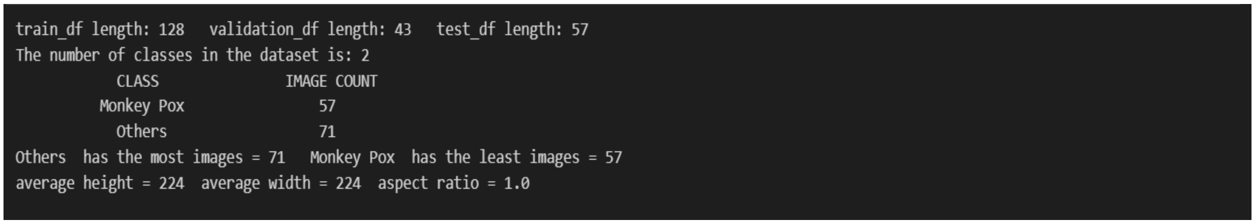


Figure 14: Samples from the Monkeypox dataset

The images highlight the visual differences that the model needs to learn to distinguish between monkeypox and other similar infections.

The results below show more details about the data I used:



```
train_df length: 128   validation_df length: 43   test_df length: 57
The number of classes in the dataset is: 2
            CLASS              IMAGE COUNT
          Monkey Pox                57
            Others                  71
Others  has the most images = 71   Monkey Pox  has the least images = 57
average height = 224  average width = 224  aspect ratio = 1.0
```

Figure 15: Data details

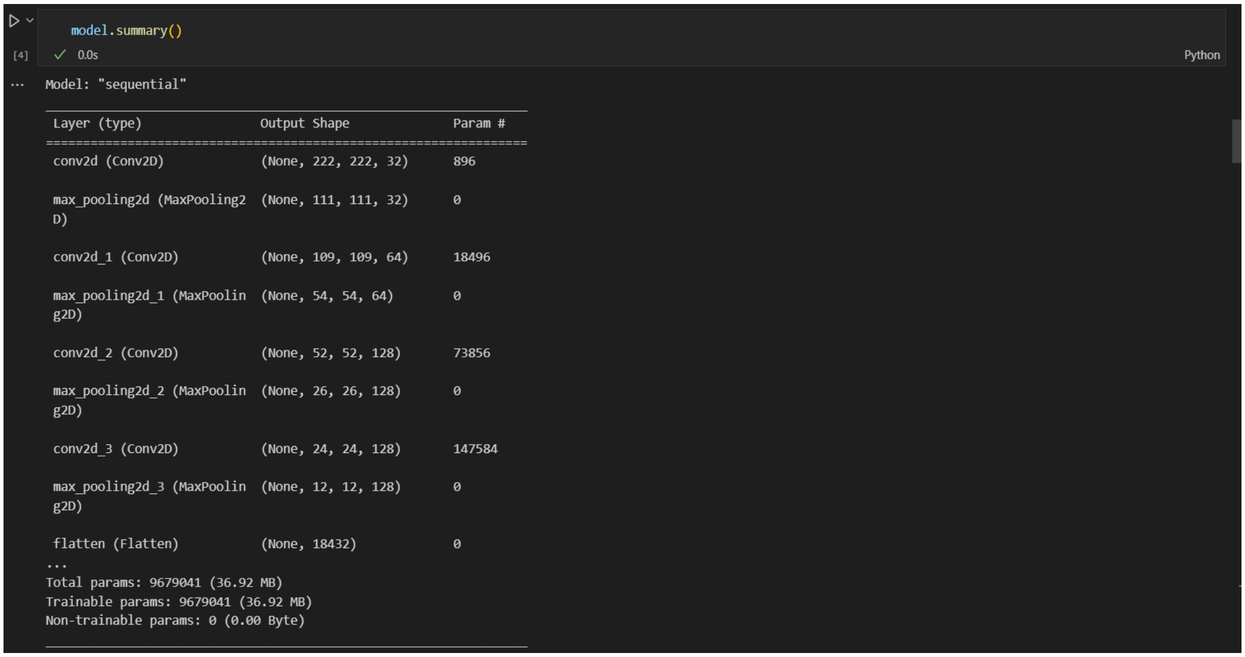
3.1.2 Building the initial ConvNet architecture

To establish a baseline, I built a simple convolutional neural network (ConvNet) from scratch using Keras. The initial architecture was designed as follows:

Table 2: Initial model architecture

Layer Type	Activation Function	Loss Function	Optimizer
Conv2D	relu	Binary Crossentropy	RMSprop with learning rate 1e-4
Dense	relu	Binary Crossentropy	RMSprop with learning rate 1e-4
Dense	sigmoid	Binary Crossentropy	RMSprop with learning rate 1e-4

The dimensions of the feature maps change with every successive layer as follows:



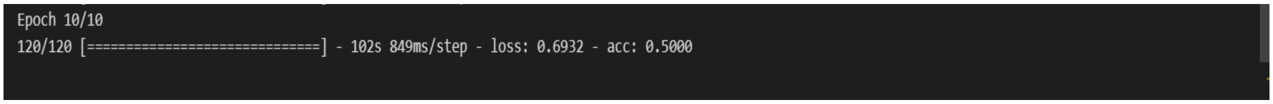
```
model.summary()
```

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv2d (Conv2D)             (None, 222, 222, 32)      896

 max_pooling2d (MaxPooling2  (None, 111, 111, 32)      0
 D)

 conv2d_1 (Conv2D)           (None, 109, 109, 64)      18496

 max_pooling2d_1 (MaxPoolin  (None, 54, 54, 64)        0
 g2D)

 conv2d_2 (Conv2D)           (None, 52, 52, 128)       73856

 max_pooling2d_2 (MaxPoolin  (None, 26, 26, 128)       0
 g2D)

 conv2d_3 (Conv2D)           (None, 24, 24, 128)       147584

 max_pooling2d_3 (MaxPoolin  (None, 12, 12, 128)       0
 g2D)

 flatten (Flatten)           (None, 18432)             0
...
Total params: 9679041 (36.92 MB)
Trainable params: 9679041 (36.92 MB)
Non-trainable params: 0 (0.00 Byte)
_________________________________________________________________
```

Figure 16: Initial architecture of the ConvNet model used for monkeypox classification

The network includes convolutional layers, max-pooling layers, and fully connected layers.Despite its simplicity, the model serves as a baseline to assess the challenges of the classification task.

3.1.3 Model perfomrmance



```
Epoch 10/10
120/120 [==============================] - 102s 849ms/step - loss: 0.6932 - acc: 0.5000
```

Figure 17: Initial model results

The initial model achieved an accuracy of 0.5, with a high loss, indicating overfitting due to the small dataset size.

- This result emphasizes the need for data augmentation and a more robust architecture to improve performance.

3.2 Model for Augmented Data (Enhanced Model)

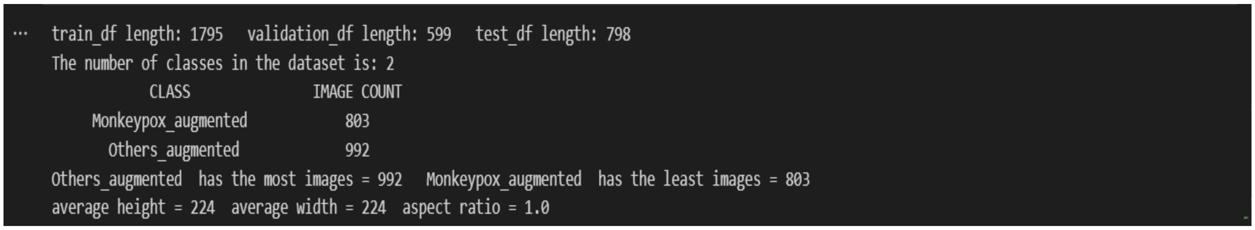
3.2.1 Data augmentation

To address overfitting and improve model performance, I applied data augmentation techniques. The images were augmented by:

- Decoding JPEG content to RGB
- Converting into floating-point tensors
- Rescaling pixel values
- Applying data augmentation

The augmented dataset was organized into:

- Training set: 1795
- Validation set: 599
- Test set: 798



```
...  train_df length: 1795   validation_df length: 599   test_df length: 798
     The number of classes in the dataset is: 2
                 CLASS                IMAGE COUNT
          Monkeypox_augmented              803
            Others_augmented               992
     Others_augmented  has the most images = 992   Monkeypox_augmented  has the least images = 803
     average height = 224  average width = 224  aspect ratio = 1.0
```

Figure 18: Augmented data organization

Samples of the augmented images (see figure 19)

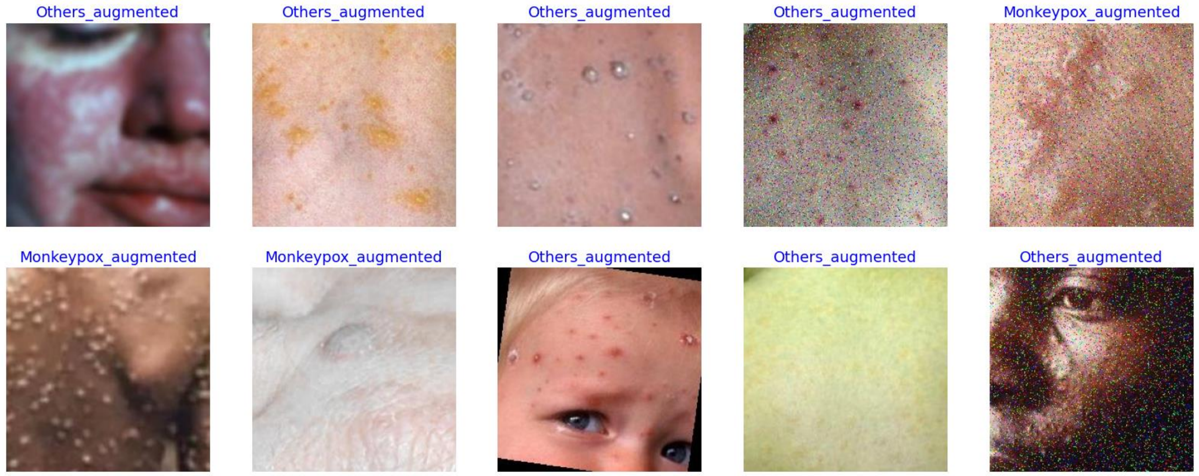


Figure 19: Data augmentation

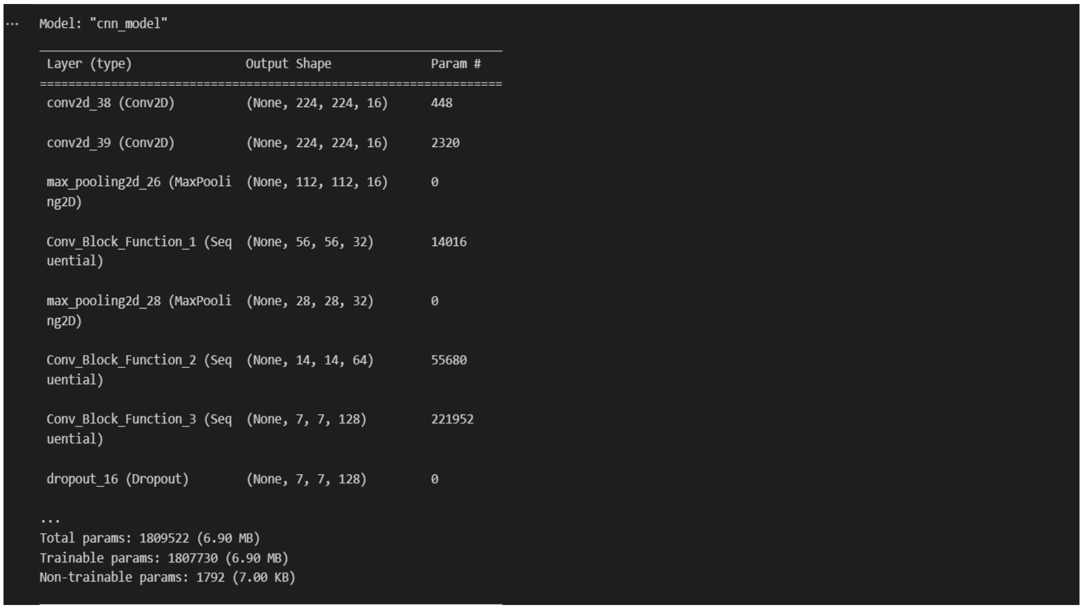
3.2.2 Building the CNN-enhanced model

With this augmented data, I designed a more complex ConvNet architecture to enhance classification performance.

- **Enhanced ConvNet Architecture:** With augmented data, I designed a more complex ConvNet architecture to improve classification performance:

Table 3: Enhanced model architecture

Layer Type	Activation Function	Loss Function	Optimizer
Conv2D	relu	Binary Crossentropy	Adam
Dense	relu	Binary Crossentropy	Adam
Dense	softmax	Binary Crossentropy	Adam

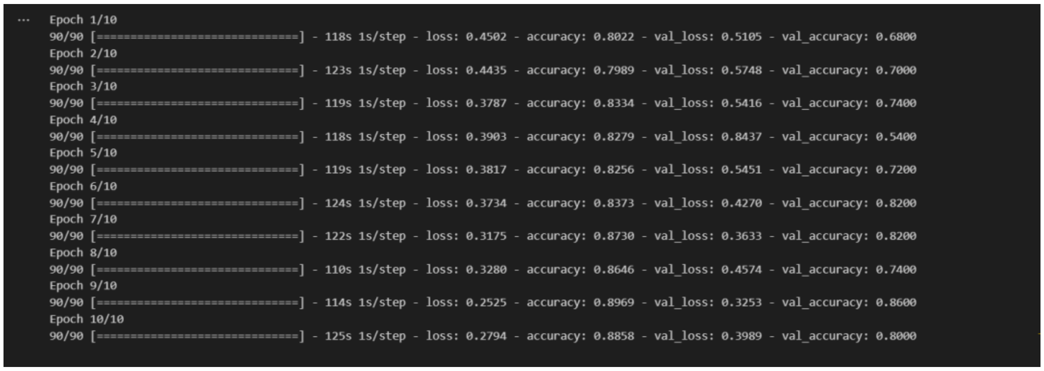


```
...  Model: "cnn_model"
     _________________________________________________________________
      Layer (type)                Output Shape              Param #
     =================================================================
      conv2d_38 (Conv2D)          (None, 224, 224, 16)      448

      conv2d_39 (Conv2D)          (None, 224, 224, 16)      2320

      max_pooling2d_26 (MaxPooli  (None, 112, 112, 16)      0
      ng2D)

      Conv_Block_Function_1 (Seq  (None, 56, 56, 32)        14016
      uential)

      max_pooling2d_28 (MaxPooli  (None, 28, 28, 32)        0
      ng2D)

      Conv_Block_Function_2 (Seq  (None, 14, 14, 64)        55680
      uential)

      Conv_Block_Function_3 (Seq  (None, 7, 7, 128)         221952
      uential)

      dropout_16 (Dropout)        (None, 7, 7, 128)         0

     ...
     Total params: 1809522 (6.90 MB)
     Trainable params: 1807730 (6.90 MB)
     Non-trainable params: 1792 (7.00 KB)
     _________________________________________________________________
```

Figure 20: Enhanced ConvNet architecture with data augmentation applied

The network includes additional convolutional layers and dropout layers to improve generalization and reduce overfitting, enabling better performance on the augmented dataset.

3.2.3 Model performance



```
...  Epoch 1/10
     90/90 [==============================] - 118s 1s/step - loss: 0.4502 - accuracy: 0.8022 - val_loss: 0.5105 - val_accuracy: 0.6800
     Epoch 2/10
     90/90 [==============================] - 123s 1s/step - loss: 0.4435 - accuracy: 0.7989 - val_loss: 0.5748 - val_accuracy: 0.7000
     Epoch 3/10
     90/90 [==============================] - 119s 1s/step - loss: 0.3787 - accuracy: 0.8334 - val_loss: 0.5416 - val_accuracy: 0.7400
     Epoch 4/10
     90/90 [==============================] - 118s 1s/step - loss: 0.3903 - accuracy: 0.8279 - val_loss: 0.8437 - val_accuracy: 0.5400
     Epoch 5/10
     90/90 [==============================] - 119s 1s/step - loss: 0.3817 - accuracy: 0.8256 - val_loss: 0.5451 - val_accuracy: 0.7200
     Epoch 6/10
     90/90 [==============================] - 124s 1s/step - loss: 0.3734 - accuracy: 0.8373 - val_loss: 0.4270 - val_accuracy: 0.8200
     Epoch 7/10
     90/90 [==============================] - 122s 1s/step - loss: 0.3175 - accuracy: 0.8730 - val_loss: 0.3633 - val_accuracy: 0.8200
     Epoch 8/10
     90/90 [==============================] - 110s 1s/step - loss: 0.3280 - accuracy: 0.8646 - val_loss: 0.4574 - val_accuracy: 0.7400
     Epoch 9/10
     90/90 [==============================] - 114s 1s/step - loss: 0.2525 - accuracy: 0.8969 - val_loss: 0.3253 - val_accuracy: 0.8600
     Epoch 10/10
     90/90 [==============================] - 125s 1s/step - loss: 0.2794 - accuracy: 0.8858 - val_loss: 0.3989 - val_accuracy: 0.8000
```

Figure 21: enhanced convnet results

- ✓ The enhanced model achieved an **accuracy of 0.8858 with a reduced loss of 0.2794**.
- ➢ The model reached an accuracy of 88%, a 30% relative improvement over the non-regularized model.

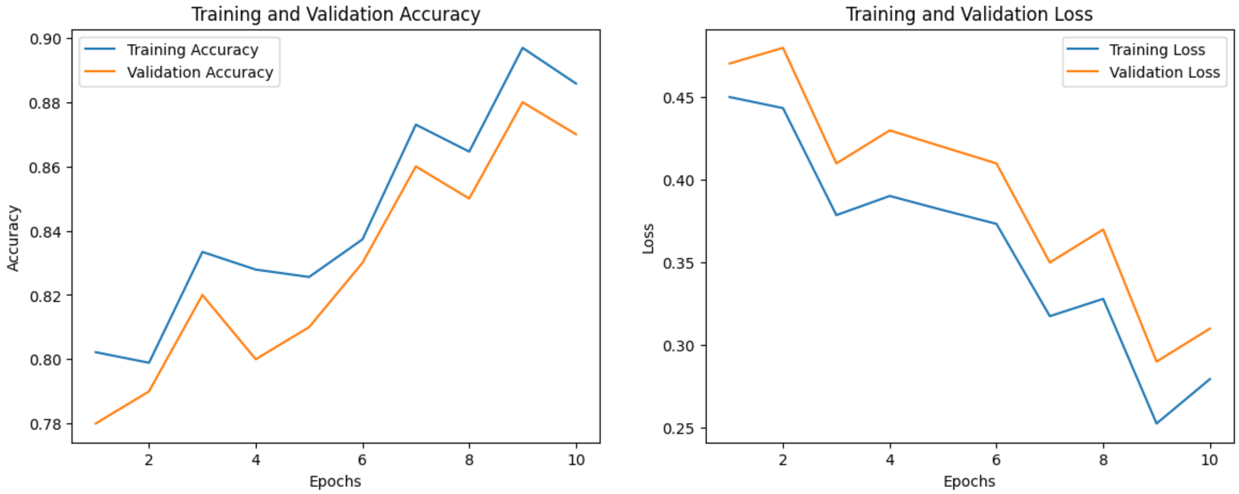


Figure 22: Curves of loss and accuracy during training

- The enhanced model, built using augmented data, demonstrated significant improvement in performance. With a significant increase in accuracy and a decrease in loss. **the training curves are closely tracking the validation curves**, which demonstrates the effectiveness of the enhanced architecture and data augmentation in overcoming the overfitting limitation of the initial model.

4- Project limitations and future perspectives

4.1 Limitations

While the current model has shown promising results, there are inherent limitations:

4.1.1 Data Scarcity

The primary challenge I encountered was the limited amount of data available. Training a deep convolutional neural network (CNN) from scratch requires a substantial amount of data to achieve optimal performance. The small dataset size has constrained the model's ability to learn intricate features and patterns.

4.1.2 Model Complexity

With a small dataset, even advanced regularization techniques might not fully address overfitting. The model might perform well on the training data but may not generalize effectively to new, unseen data.

4.2 Future Perspectives

Despite achieving a high accuracy with the current model, there is potential for further improvement. To enhance the model's performance, several advanced techniques can be explored:

4.2.1 Transfer Learning

- Pretrained Models:

Leveraging pre-trained convolutional neural networks (CNNs) such as VGG, ResNet, or Inception as feature extractors. These models, trained on large datasets, can provide a solid starting point and improve performance when fine-tuned on the monkeypox dataset.

- Fine-Tuning:

Adjusting the number of layers and the learning rates for these pre-trained models to adapt them specifically to the monkeypox classification task.

4.2.2 Network Architecture Optimization

- Hyperparameter Tuning:

Experimenting with different numbers of filters in convolutional layers, varying kernel sizes, and exploring deeper or shallower network architectures.

References

1. **NCBI GEO Database**: For the raw data used in the analysis.

    URL: https://www.ncbi.nlm.nih.gov/geo/

2. **DESeq2 (R package)**: Used for differential expression analysis.

    Reference: Love MI, Huber W, Anders S. "Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2." Genome Biology, 2014.

3. **Enrichr**: Used for gene set enrichment analysis.

    Reference: Chen EY, Tan CM, Kou Y, et al. "Enrichr: interactive and collaborative HTML5 gene list enrichment analysis tool." BMC Bioinformatics, 2013.

    URL: https://maayanlab.cloud/Enrichr/

4. **WEB-based Gene Set Analysis Toolkit**: Used for functional annotation.

    Reference: Wang J, Vasaikar S, Shi Z, Greer M, Zhang B. "WebGestalt 2017: a more comprehensive, powerful, flexible, and interactive gene set enrichment analysis toolkit." Nucleic Acids Research, 2017.

    URL: http://www.webgestalt.org/

5. **Cytoscape**: For network visualization and hub gene identification.

    Reference: Shannon P, Markiel A, Ozier O, et al. "Cytoscape: a software environment for integrated models of biomolecular interaction networks." Genome Research, 2003.

    URL: https://cytoscape.org/

6. **CytoHubba**: Cytoscape plugin used for hub gene identification.

    Reference: Chin CH, Chen SH, Wu HH, Ho CW, Ko MT, Lin CY. "cytoHubba: identifying hub objects and sub-networks from complex interactome." BMC Systems Biology, 2014.

7. **Kaggle**: Used for sourcing the image data for the convolutional neural network (CNN) model.

    URL: https://www.kaggle.com/

8. **Keras**: Used for building the convolutional neural network (CNN) models.

    Reference: Chollet F. "Keras: The Python Deep Learning library." 2015.

    URL: https://keras.io/