

**Student ID: 3810497**  
**Name: Subham Kumar**  
**Rahman**

**Student: 3825159**  
**Name: Kowsar**

## **Abstract**

Our goal was to explore various deep neural networks for the detection and classification of various cell types in the histopathology images of cancer cells caused due to colon cancer. Our exploration and research included two types of **Convolutional Neural Networks (CNN): Very Deep Convolutional Networks (VGG)** for Cancer Detection and **both Residual Networks (ResNet) and VGG** for cell type classification with improvisation and optimization to provide accurate results.

The dataset consisted of 27x27 RGB images of cancer cells from 99 different patients. While inspecting the data, information from **patientID 76** was missing so data of 98 patients were used.

## **Exploratory Data Analysis**

The dataset consists of **20280 images** that have been classified as cancerous or non cancerous. Out of those, only **9896 images have a label for cell-type** named as fibroblast, inflammatory, epithelial, and others, whereas **10384 images do not have a label for cell type**.

According to the class distribution output and the column chart displayed in **Figure 1(Appendix)**

It can be seen that about **4079 of the images are epithelial (41.22%)**, **2543 as inflammatory (25.77%)**, **1888 as fibroblast (19.01%)** and **1386 belong to the others category (14%)**.

**Figure 2(Appendix)** shows the 4 different cell types as digits: **(0 for fibroblast, 1 for inflammatory, 2 for epithelial, 3 for others)** while **Figure 3(Appendix)** shows the relationship between the cell types and if they are cancerous or not. We can clearly see from Figure 3 that **only epithelial cells are cancerous**.

## **Data Preprocessing and Data Augmentation**

### **For Task 1:**

Using the **'train\_test\_split'** function, the dataset is divided into three subsets which are: training, testing and validation. 20 percent of the data would be used for testing while 80 percent of the data is to be used for training. To get a better understanding of the neural networks, we used the Sequential Model API given by TensorFlow.

For the **baseline model**, we used the VGG Model of 3 blocks. The VGG Model may be computationally expensive but it is simpler to implement than other models. It follows an uniform pattern and consists of repeated blocks of convolutional layers, followed by max pooling layers, which makes it easy to understand and implement.

Before working on the final model, we again worked on **data augmentation**. This time we made some edits to our existing images. We rotated our images to 20 degrees and gave a brightness range from **0.2 to 1.2**. In this model, we didn't use any dropout. The results of this model were so far the best in terms of accuracy. It gave an accuracy of 0.9 which is exactly the same as the original research paper published and also produced a significant decrease in the test loss which recorded 0.26. A lower test loss indicates that the model's predictions are closer to the true labels, indicating better performance. A lower test loss implies that the model has effectively learned the patterns and features in the training data and can generalize well to unseen test data. The final model also had a F1 score of 0.9 too as seen by the learning curve in **Figure 6(Appendix)**. Both the second and third model were saved so that they can be tested independently. Hence, we decided to go with the VGG\_Model\_3 and carry our prediction using image visualization. As shown in **Figure 10(Appendix)**, any image input from the patch\_images directory can be entered into the code and the model will predict if it is cancerous or not.

### **For Task 2:**

Similar to Task 1, the dataset is divided using train\_test\_split with a 80-20 split. The labels are converted to string format before the ImageDataGenerator is loading the images in batches to memory to be able to compile the baseline model.

Once two baseline models (VGG and ResNet) have been compared, **data augmentation has been used** with a rotation range of 20, width shift and height shift range of 0.5 and a brightness range of 0.5 to 1.5 for improvised results using increased diversity from such a large dataset of images before using regularization.

## **Experimentation**

### **For Task 1:**

For the baseline model, it contains two convolutional layers followed by a max pooling layer. The model also contains a flatten layer. The baseline model gave an accuracy of 0.87 and a test loss of 0.47 as seen in **Figure 4(Appendix)**.

The baseline model was underfitting hence in the next model, we decided to go for L2 regularization and a dropout of 0.2. The accuracy of that model was 0.89 and there was a test loss of 0.33 which can be observed by the learning curve in **Figure 5(Appendix)**. This meant that we were indeed going close to our goal of the original model which had an accuracy of 0.9.

## **For Task 2:**

As shown in **Figure 7(Appendix)**, the baseline models for both VGG and ResNet were overfitting however VGG had worked better than ResNet providing less test loss and more accuracy. **The test loss and accuracy for the baseline VGG was 270.6% and 66.56% while baseline ResNet had a test loss of 422.1% and an accuracy of 69.2%.** Hence, the baseline model was chosen to be augmented and regularized which **reduced the test loss to 77.56% and improved the accuracy to 71.56%** as proven by the learning curve in **Figure 8(Appendix)**.

When deciding to further improve the model using **dropout**, the performance was reduced slightly with an **increased test loss to 80.3%** and **reduced accuracy to 70.25%**. However, this model was still decided to be used on `data_labels_extraData.csv` to see if it performs well on the testset. Unfortunately, the performance was even worse as it **increased the test loss to 96.4%** and **reduced the accuracy to 67.1%**. The prediction scores for all the models can be seen in **Figure 9(Appendix)**.

## **Ultimate Judgement of the Report**

The model achieved an impressive **accuracy of 90.86%** and **F1 score of 90.5%** on the test dataset for Task 1 with a **training loss of 26% only**, indicating its ability to correctly classify the majority of cell samples. This high accuracy suggests that the VGG model has learned meaningful representations and features from the input images, enabling it to discern between cancerous and non-cancerous cells with a remarkable degree of accuracy after regularization and data augmentation.

On other hand, the **VGG model seems to be performing better than ResNet in task 2**, and performs even better when augmented and regularized. However, the performance is lowered when applying dropout and when tested with unlabeled data from `data_labels_extraData.csv`. The regularization model provides a good **accuracy of 71.56%** and **F1 score of 64.2%** which is **reduced to 70.25% and 60.8%** respectively when dropout has been applied but somehow slightly improves the precision of the model. Hence, the **augmented and regularized VGG model seems to outperform** every other experimented model in this task.

## **Conclusion**

The VGG model employed in this study has demonstrated significant potential in accurately predicting whether a cell is cancerous or not with its cell type classifications. Through rigorous evaluation and analysis, the model has consistently exhibited strong performance, providing reliable predictions for the classification of cancer cells task at hand even though it faced severe test loss during the cell type classification.

Appendix:

Image sizes: {(27, 27)}  
Image channels: {3}  
Class distribution:  
epithelial 4079  
inflammatory 2543  
fibroblast 1888  
others 1386  
Name: cellTypeName, dtype: int64  
Mean pixel value: [193.56498092 150.12985821 205.31894543]  
Std dev of pixel values: [38.15567203 41.74090031 27.33528803]  
Outliers: []

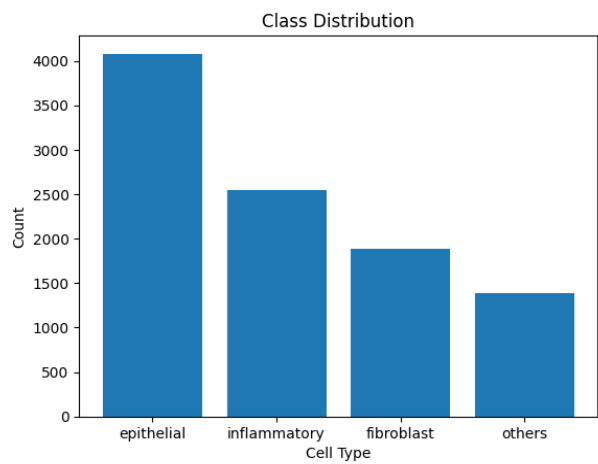


Figure 1: EDA for Cell Type Classification

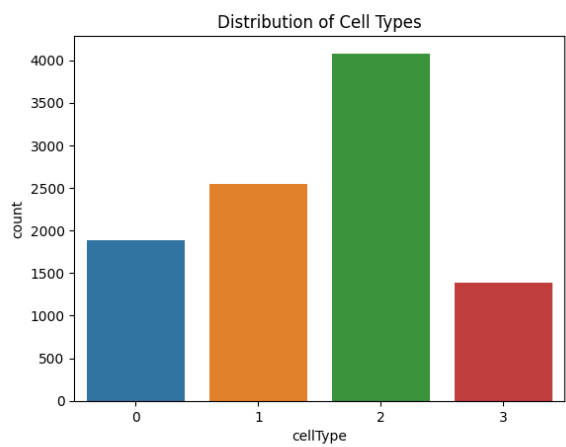


Figure 2: Cell type classification in numbers

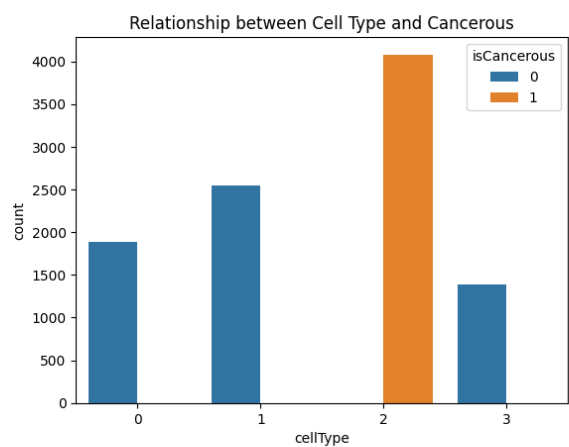


Figure 3: cellType relationship to isCancerous

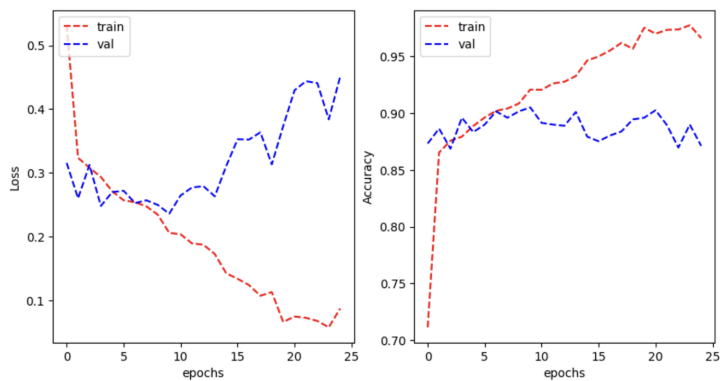
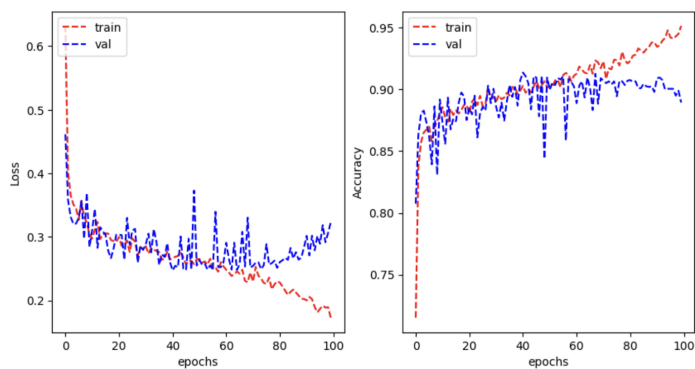
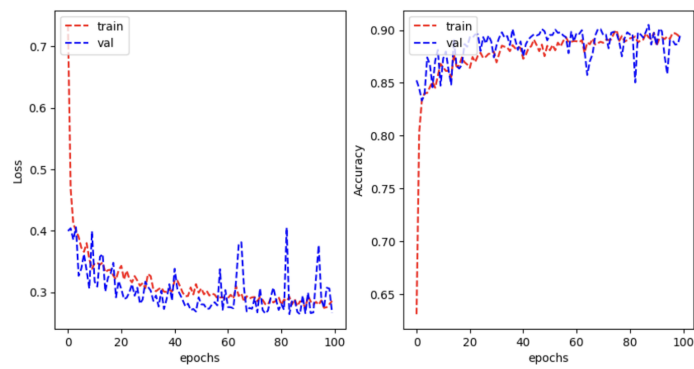


Figure 4: Learning curve of baseline model for Task 1



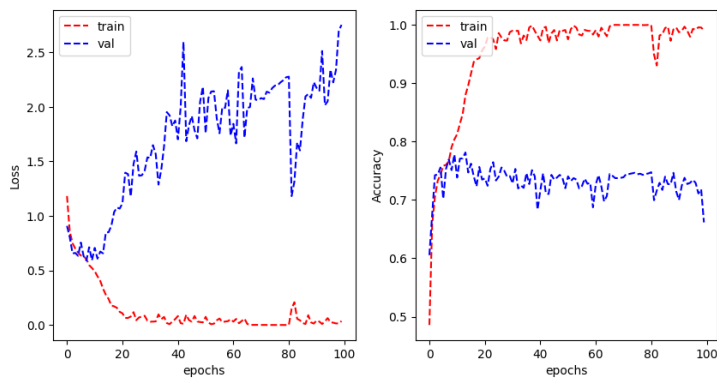
**Figure 5:** Learning curve of VGG Model 2

3

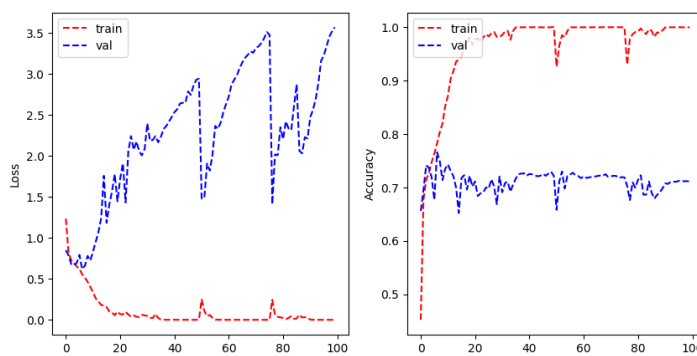


**Figure 6:** Learning curve of VGG Model

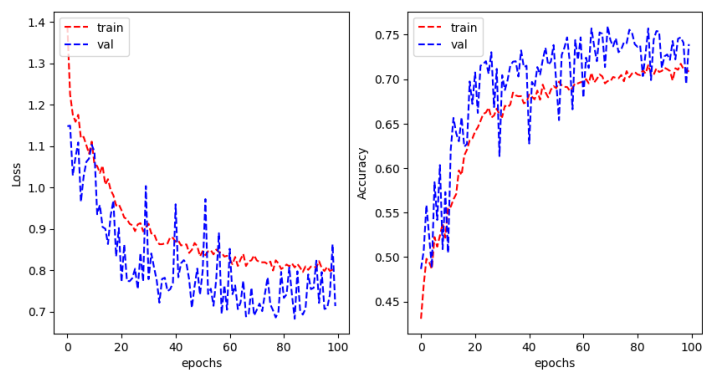
### VGG for Task 2



### ResNet for Task 2



**Figure 7:** Comparison of test loss and accuracy scores for baseline models for Task 2



**Figure 8: Task 2 VGG model learning curve after data augmentation and regularization**

**VGG Model augmented and regularized**

```

Accuracy score: 0.7156565656565657
F1 score: 0.6420704158145139
[[160 73 40 118]
 [ 5 418 20 53]
 [ 17 58 713 3]
 [ 17 118 41 126]]

```

	precision	recall	f1-score	support
0	0.80	0.41	0.54	391
1	0.63	0.84	0.72	496
2	0.88	0.90	0.89	791
3	0.42	0.42	0.42	302
accuracy			0.72	1980
macro avg	0.68	0.64	0.64	1980
weighted avg	0.73	0.72	0.71	1980

**VGG model after adding dropout**

```

Accuracy score: 0.7025252525252526
F1 score: 0.6068852524737146
[[155 147 62 27]
 [ 9 457 21 9]
 [ 4 77 710 0]
 [ 16 175 42 69]]

```

	precision	recall	f1-score	support
0	0.84	0.40	0.54	391
1	0.53	0.92	0.68	496
2	0.85	0.90	0.87	791
3	0.66	0.23	0.34	302
accuracy			0.70	1980
macro avg	0.72	0.61	0.61	1980
weighted avg	0.74	0.70	0.68	1980

**VGG model tested on unlabelled data from extraData.csv**

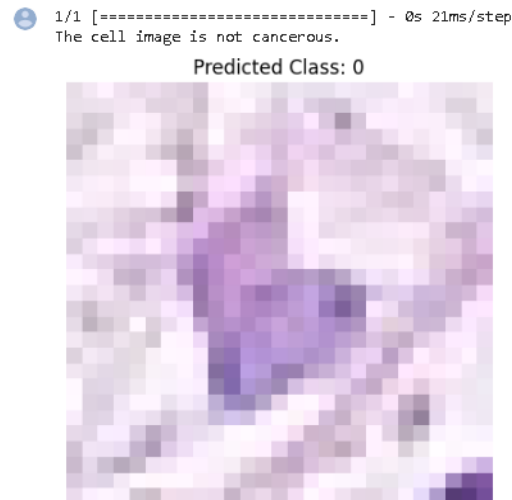
```

Accuracy score: 0.6707070707070707
F1 score: 0.5971890889681217
[[135 83 75 98]
 [ 1 435 15 45]
 [ 2 135 650 4]
 [ 11 151 32 108]]

```

	precision	recall	f1-score	support
0	0.91	0.35	0.50	391
1	0.54	0.88	0.67	496
2	0.84	0.82	0.83	791
3	0.42	0.36	0.39	302
accuracy			0.67	1980
macro avg	0.68	0.60	0.60	1980
weighted avg	0.72	0.67	0.66	1980

**Figure 9: Test scores for all VGG models tested in Task 2**



**Figure 10: Task 1 predicting if the cell image is cancerous or not**

EDA: [https://colab.research.google.com/drive/1PzUoC9GUYyggH8hJy0ZsXfw0\\_M3Efp0#scrollTo=xIW-x2EnB2cV](https://colab.research.google.com/drive/1PzUoC9GUYyggH8hJy0ZsXfw0_M3Efp0#scrollTo=xIW-x2EnB2cV)  
Task1CancerDetection: [https://colab.research.google.com/drive/1riO\\_x31qlor-Xye4wTg05LxpSZ257QPc?authuser=1#scrollTo=3W--W1Li3B9U](https://colab.research.google.com/drive/1riO_x31qlor-Xye4wTg05LxpSZ257QPc?authuser=1#scrollTo=3W--W1Li3B9U)  
Task2CellTypeClassification: <https://colab.research.google.com/drive/1aoozqTSzLsyCeL76Wehikbja9WeQuvLe?authuser=8#scrollTo=exyasNca3dQT>