SKIN LESION ANALYSIS TOWARDS MELANOMA DETECTION

Abhinay Gupta Somisetty
Master's in CSC
California State Univrsity,Sacramento
asomisetty@csus.edu

Harshit Rajesh Ambalkar
Master's in CSC
California State University, Sacramento
hambalkar@csus.edu

Abstract—

Machine learning is becoming a helping hand in many domains and so it has also gained some fame in the medical field. Melanoma is one of the major causes of skin cancer and early detection of melanoma in dermatoscopic images can save many lives. Reliable automatic classification of skin lesions can help pathologists improve their efficiency.

DCNN has been used in many other fields and has further scope for research in other domains, Recently, by transfer learning [1], Esteva et al proposed in "Dermatologist – level classification of Skin Cancer with Deep Neural Networks" that "CNN achieves performance on par with all tested experts, demonstrating an artificial intelligence capable of classifying skin cancer with a level of competence comparable to dermatologists". Inspired by his work, We have attempted to apply different Deep Convolutional Neural Network models that have succeeded with the ImageNet dataset for classifying 7 different types of skin lesions. We also built a basic CNN model to get the difficulty of classifying skin lesions. Fine-tuning the top layers were performed with Inception V3 and Densenet201. Densenet201 performed the best giving 83% accuracy on the test set, even though Densenet201 has fewer parameters then IncpetionV3. Results of the experiment verify that the features learned by pretrained models and architecture of DCNNs help to learn features of a completely different domain, here which is a skin lesion dermatoscopic dataset.

Keywords: Lesion, CNN, Classification, feature, Melanoma.

I. INTRODUCTION

Skin cancer is the most common diseases with 5 million people effected every year in United States. There are many diseases causing skin cancer. Not all the skin lesions lead to skin cancer. So these skin lesions are divided into malignant and benign. Cancer causing lesions are distinguished as malignant and harmless lesions are benign. In recent times there are lot of research carried out in machine learning to classify the cancer causing lesions which helps medical industry to early classify the cancer causing lesions and proper medication is given to the patient at the early stage. Lesion classification involves image classification which means the lesion image is given as input to any model and classify it if it is benign or malignant. The dataset consists of 10015 dermoscopic images that can serve as a benchmark for skin lesion diagnosis. HAM was used as a training set for the ISIC 2018 Skin Lesion Diagnosis Challenge: Skin lesson analysis towards Melanoma detection grand challenge datasets[3],which consists of dermoscopic lesion images in JPEG format. It consists of 10015 images(327 actinic keratosis,514 basal cell carcinoma,115 dermatofibroma, 1113 melanoma, 6705 nevus, 1099 pigmented benign keratosis, 142 vascular lesions)All lesion images are named using the scheme ISIC_<image_id>.jpg, where <image_id> is a 7-digit unique identifier.

Metadata summary

Design Type(s)	• Database creation objective • Data integration objective • Image format conversion objective
Measurament type(s)	• Skin lesions
Technology Types	• Digital curation
Factor Types(s)	• Diagnosis • Diagnostic procedure • Age • Biological sex • Animal body part
Sample characteristics	• Homo sapiens • Skin of body

We will be using 70% of data for the training set 10% for validation and 20% for the test set, applying 3 folds cross validation to get best hyper parameter value for the model. [4]The dataset contains unequal distribution of malignant and benign lesions, having more benign lesions then malignant lesion, the distribution of disease states represent a modified "real world" setting. The project flow goes as follows.

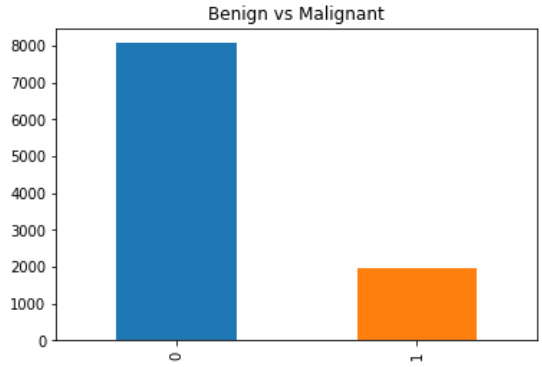
- Data exploration
- Base model implementation
- Implementation of InceptioV3
- Implementation of DenseNet201

II. APPROACH

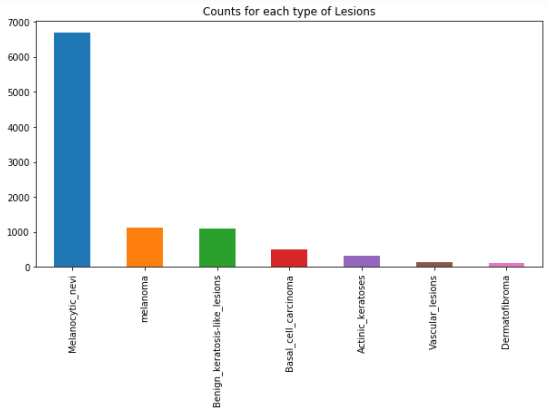
A. Exploring the Dataset

Exploring the data also involves preparation for training the CNN models as well. In the given dataset 2000 examples are malignant and 8000 examples are benign. Melanoma and Basal cell carcinoma and Actinic keratosis are cancer causing.

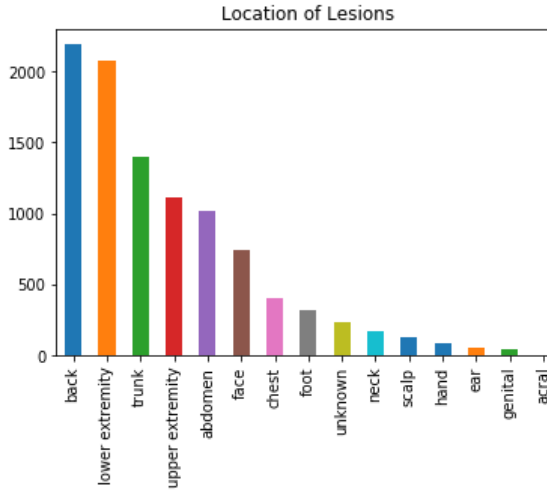
- We have declared the dangerous diseases which causes cancer and we have counted and plotted the malignant vs Benign. Below is the graph.



- Now we are counting for each type of Lesion in our dataset and plotting the graph between the lesions. We see that dataset has more samples of melanocytic_nevi lesion which is benign.

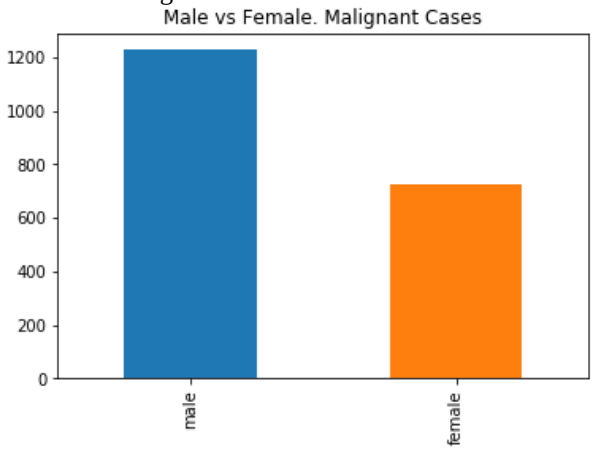


- We also have location of the lesion present in the body. Below is the graph plotted where lesions are present for each location in the body.

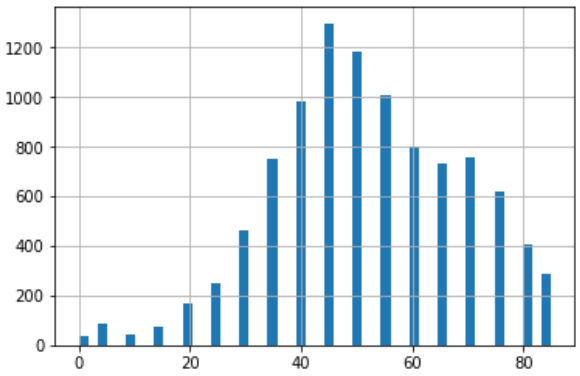


Some of the characteristics of the patient samples are discussed below.

- We have also differentiated the data of the samples which are malignant in nature based on the sex.

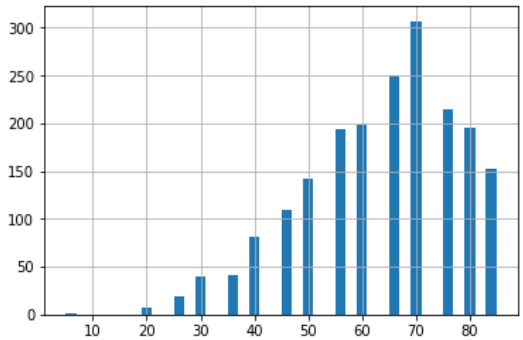


- Graph plotted for the samples based on the age of the patient.



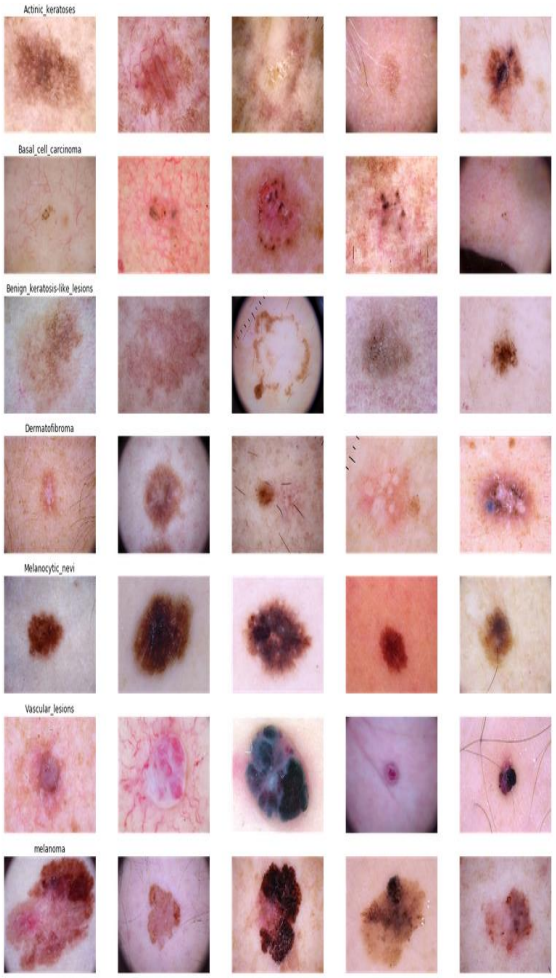
From the above graph we observe that the most of the sample ages more than 30.

- Now we will plot the graph for age of the samples which are malignant in nature.

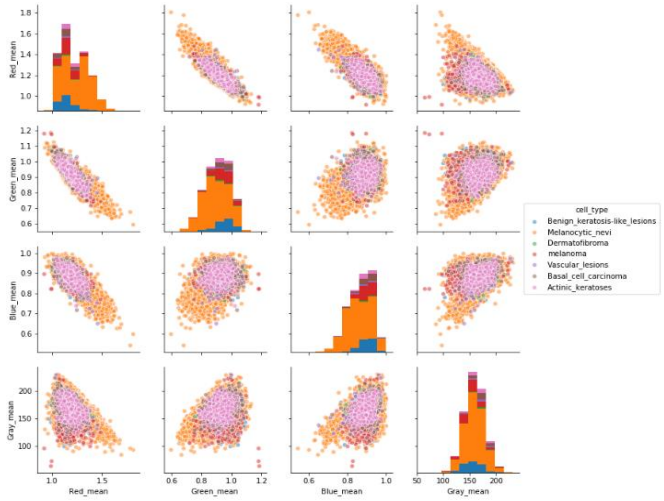


From the above graph we can say that the ages of sample with malignant nature is high for patients with age more than 50.

- 5 sample images are plotted for each type.



- The shape of the image array is 450, 600, 3. Where 3 is the image channel. Which are nothing but Red, Blue and Green. We find the mean of each colors and draw a plot of lesions over these colors.



- Now we have plotted the random sample images of each lesion
- Now in order to give input to the models we have resized the images in the dataset.
    1. For the baseline model we have resized the images to 64*64 pixels and created a data frame and saved it in a CSV file for the future use.
    2. For InceptionV3 and DenseNet models we have created a standardized resized images of size 221*221*3 to give as a input. The resized images are split into train, validation and test datasets and save in numpy objects for future use.

III. Implementation of Model

In the implementation part we are building our own model as well as we are implementing transfer learning and training the models like inception v3 and densenet201 on our dataset and evaluating the results which includes accuracy and loss calculated by the above mentioned models on the discussed dataset.

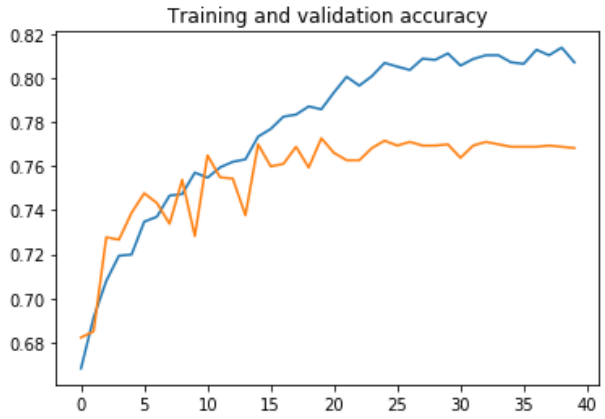
For Baseline model we have used input dataset of size 64*64*3. We have also resized the actual dataset to a standardized dataset of 221*221*3 to give as input to the Inception and DenseNet models.

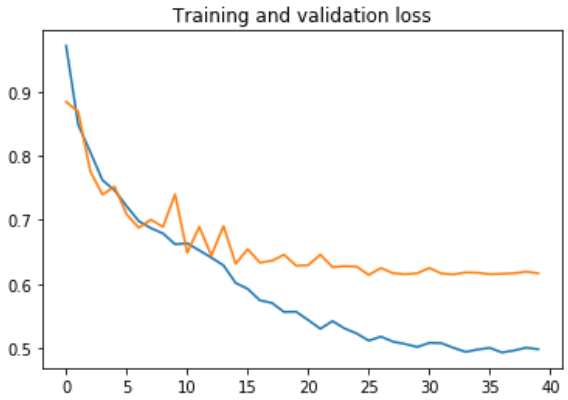
A. Baseline

A small CNN baseline model is developed to check the accuracy before we implemented the actual CNN models. We have resized the images to 64*64 pixels to give as input for this model.

- First a 16 kernels convolutional layer that are 3*3 and padding is maintained same.
- Next, we have applied a max-pooling layer of 2*2 and the result is the feature map which is reduced by 2.
- Again, we have used 32 kernels convolutional layer that are 3*3 and max pooling is applied of 2*2 layer.
- Similarly with 64 kernels.

We used Dense layer with a 512 hidden layers and Relu activation function . Dropout function is used which is a regularization technique with the value of 0.5 to tackle the overfitting. We used Adam optimizer which calculates an exponential moving average of the gradient and the squared gradient, and the parameters beta1 and beta2 control the decay rates of these moving averages with learning rate of 0.001. A loss function Categorical cross entropy is used for single label categorization. This is when only one category is applicable for each data point. We used ImageDataGenerator for data augmentation with a rotation range of 40 and horizontal shift range of 0.2 which means the data is no longer static and it keeps on changing. ReduceLROnPlateau, a learning rate reducing adaptive method is used to reduce the learning rate when the metric stopped improving. The target metric we used is ‘val_acc’ with factor=0.5 and patience is 3. We have declared batch size as 64 and epoch as 30. To train out model we are using fit_generator function[3] which takes input params like train data, number of epochs declared above, validation data, steps per epoch which tell the model when to exit and start new epoch. Now the above trained model is evaluated on both validation data and test data and accuracy and loss is evaluated. Now, a graph is plotted between training vs validation accuracy as well as training vs validation loss.





B. Inception V3

Now using transfer learning we are implementing InceptionV3 a pre-trained model on imagenet weights for which we are giving input_size as 221,221, 3. The standardized input.

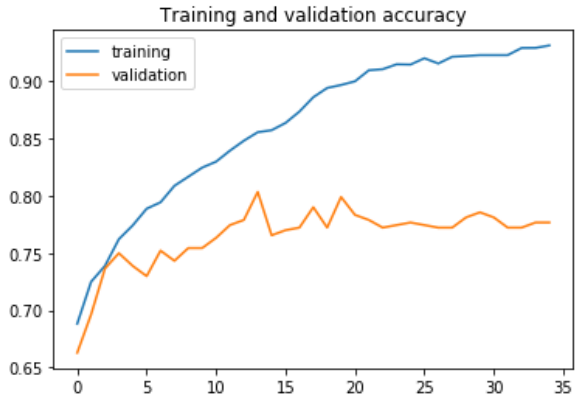
- Training:

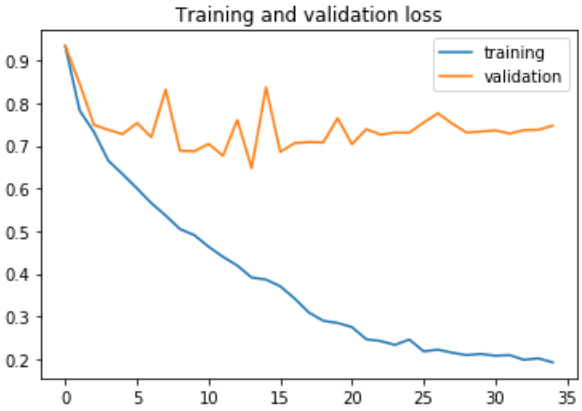
  We will fetch the layers for the given input size and on the last_output layer of the pre trained model we perform global max pooling and use adam optimizer. Then find out the trainable and non-trainable parameters. We observed the number of non-trainable parameters are very high than the trainable parameters. Now did data augmentation and trained the model with batch size 64 and for 3 epoch. We can see the accuracy and loss of this model.

- Fine Training:

  Now we randomly selected one layer (in our case 299) and use Adam optimizer to optimize the model and observed the summary of the model. This way has improved the number of trainable parameters to a great extent compared to previous summary. Now again the model is trained with a batch size of 64 and 35 epochs with learning rate reduce function and find the accuracy and loss of this model. Accuracy is relatively high and loss is less compared with the two models. Now, we also evaluated the model on validation and test data and accuracy and loss is calculated.

We observed that the accuracy has improved from training model to fine training model. A graph is plot between training vs validation accuracy as well as training vs validation loss.





To achieve better results than these two models we also tried to implement DenseNet201 using transfer learning technique.

C. DenseNet201

Now using transfer learning we are implementing DenseNet a pre-trained model on ImageNet weights for which we are giving input_size is 221,221, 3.

- Training:

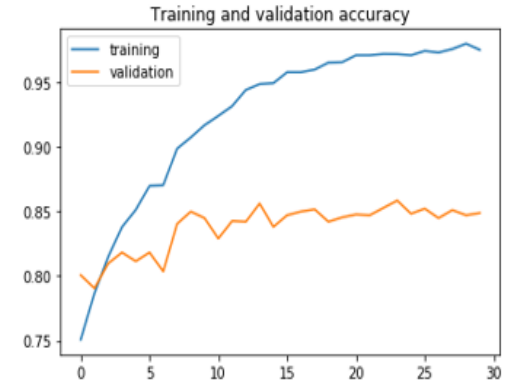
  We will fetch the layers for the given input size and on the last output layer of the pre trained model we perform global max pooling and use adam optimizer. Then find out the trainable and non-trainable parameters. We observed the number of non-trainable parameters are very high than the trainable parameters. Now did data augmentation and trained the model with batch size 16 and for 3 epoch. We can see the accuracy and loss of this model.

- Fine Training:

  Now we randomly selected one layer (in our case 481) and use Adam optimizer to optimize the model and observed the summary of the model. This way has improved the number of trainable parameters to a great extent compared to previous summary. Now again the model is trained with a

batch size of 16 and 30 epochs with learning rate reduce function and find the accuracy and loss of this model. Accuracy is relatively high and loss is less compared with the two models. Now, we also evaluated the model on validation and test data and accuracy and loss is calculated.

We observed that the accuracy has improved from training model to fine training model. A graph is plot between training vs validation accuracy as well as training vs validation loss.



So when we compare the three models, DenseNet201 has given the better accuracy. The results of the three models are discussed below in section IV.

D. Ensemble Model

Esteva et al. showed in [1] that CNNs can outperform a human expert in a classification task after an exhausted learning phase on a huge annotated training set. However, in many cases, a sufficient amount of annotated images (ground-truth) is not available, so we should improve the accuracy by other approaches.

We applied ensemble learning by ensemble two models inception V3 and Densenet giving us better results.

IV Results

Below we will discuss the results of the different models and compare the results of the different models we implemented in section III.

A. Result of Baseline Model

Metric	Dataset		
	Training	***Validation***	***Test***
Accuracy	0.8061	0.768164	0.760479
Loss	0.5012	0.616828	0.664191

B. Results of Inception V3

Metric	Dataset		
	Training	***Validation***	***Test***
Accuracy	0.9316	0.805987	0.823353
Loss	0.1925	0.67433	0.710989

C. Results of DenseNet201

Metric	Dataset		
	Training	***Validation***	***Test***
Accuracy	0. 9752	0.849224	0.837325
Loss	0.0711	0.625440	0.656559

D. Combined result for three models.

Model	Metrics		
	Accuracy	***Loss***	***Parameters***
Baseline	0.760479	0. 664191	2,24,839
InceptionV3	0.823353	0.710989	22,855,463
DenseNet201	0.837325	0.656559	19,309,127
Ensemble Model(Inception V3 + DenseNet2)	0.86	0.456307	-

V Conclusion

Using Deep Convolutional Neural Network models on HAM1000 Dataset we came to know that features learnt by these pretrained models with the help of DCNN' architecture helps learning features of different domain. Densenet having less parameter compared to Inception V3 is still able to give better accuracy then Inception V3, since Densenet receives all preceding layers as Inputs which tends to have richer patterns, Also by using ensemble learning we tend to increase the accuracy by 3 %.

VI Discussion and Future Scope

In our current dataset we have less number of malignant cases then benign. To balance the dataset we could use resampling

techniques which would give better performance, we would Also retrain our pretrained model to give better accuracy by increasing epoch.

REFERENCES

[1] Esteva, A., Kuprel, B., Novoa, R., Ko, J., Swetter, S., Blau, H., and Thrun, S. (2017). Dermatologistlevel classification of skin cancer with deep neural networks. Nature, 542(7639):115–118J. Clerk Maxwell, A Treatise on Electricity and Magnetism, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68–73.

[2] https://www.pyimagesearch.com/2018/12/24/how-to-use-keras-fit-and-fit_generator-a-hands-on-tutorial/

[3] Noel Codella, Veronica Rotemberg, Philipp Tschandl, M. Emre Celebi, Stephen Dusza, David Gutman, Brian Helba, Aadi Kalloo, Konstantinos Liopyris, Michael Marchetti, Harald Kittler, Allan Halpern: "Skin Lesion Analysis Toward Melanoma Detection 2018: A Challenge Hosted by the International Skin Imaging Collaboration (ISIC)", 2018; https://arxiv.org/abs/1902.03368

[4] Tschandl, P., Rosendahl, C. & Kittler, H. The HAM10000 dataset, a large collection of multi-source dermatoscopic images of common pigmented skin lesions. *Sci. Data* **5**, 180161 doi:10.1038/sdata.2018.161 (2018).