**MACHINE LEARNING PROJECT**

**Skin Lesions Classification using Computer Vision and Convolutional Neural Networks**

**HARSHIT NARASAPPA**

**210473140**

**Introduction:**

Computer vision-based skin cancer detection has the potential to be very useful in the identification of skin cancer. Many lives can be saved if cancer is detected and treated early. To reliably classify skin lesions as skin cancer, it is critical to have efficient and improved image processing tools. Skin cancer is a condition caused by changes in the qualities of normal skin cells that cause them to become malignant, meaning that the cells continue to divide uncontrollably owing to DNA damage. Skin cancer has an uneven structure with cell differentiation at multiple levels of chromatin, nucleus, and cytoplasm, according to histopathology. Despite the fact that the incidence of Malignant Melanoma is lower than that of Basal Cell Carcinoma and Squamous Cell Carcinoma, the death rate is higher, accounting for 75 percent of all skin cancer deaths. Convolutional Neural Networks (CNN) are being used in the research to accurately categories pigmented skin lesions in dermoscopic pictures in order to detect malignant skin lesions as early as feasible. Convolutional neural networks (CNNs) are a type of deep neural network that uses convolution instead of conventional matrix multiplication in at least one layer. They excel at assessing visual images because they are fully-connected (FC) feed-forward networks, which efficiently minimize the number of parameters without sacrificing model quality. The study looks at two convolutional neural networks with different architecture and/or depth, as well as data pre-processing methods, to evaluate how they affect skin lesion classification performance. The CNN architecture ResNet50 and ResNet152V2 models were employed.

**ResNet50**: ResNet-50 is a 50-layer deep convolutional neural network. You can import a pertained version of the network from the ImageNet database, which has been trained on over a million photos. The network can classify photos into 1000 different object categories, including keyboards, mice, pencils, and a variety of animals. As a result, the network has learned a variety of rich feature representations for a variety of images. The network's picture input size is 224 × 224 pixels. The top-5 test accuracy is 92.7%

**ResNet152V2**: The ResNet152V2 model is followed by a reshape layer, a flatten layer, a dense layer with 128 neurons, a dropout layer, and finally a dense layer with Softmax activation function to categories the picture into its appropriate class in the model architecture. The top-5 test accuracy is 94.3%

**Dataset:**

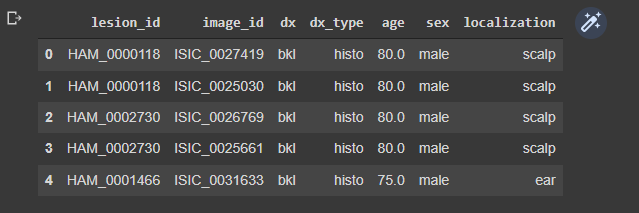
The goal of this project is to create the best accurate machine learning model for the HAM10000 dataset in order to categories skin lesion categories in photographs ("Human against Machine with 10000 training images"). This will assist clinicians in immediately identifying high-priority patients and expediting their workflow. The dataset consists of 10015 dermatoscopic images that were provided as a training set for academic machine learning and are freely accessible via the ISIC archive. (https://www.kaggle.com/kmader/skin-cancer-mnist-ham10000).There are seven attributes connected with each image and patient in the dataset:

1. lesion\_id [lesion\_id]
2. unique image\_id [image\_id]
3. diagnostic skin lesion category [dx] [To be predicted in our tests]
4. technical validation field type, which indicates how the skin lesion diagnosis was made [dx\_type]
5. patient’s age [age]
6. patient’s sex [sex]
7. localization of the skin lesion [localization]

**The following are the seven different diagnostic skin lesion classifications that must be predicted:**

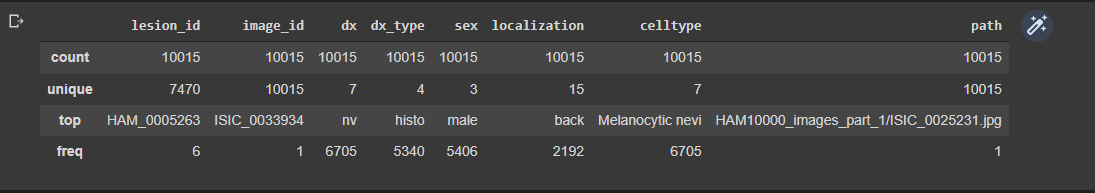
1. **Melanocytic Nevi [nv]** are benign melanocyte neoplasms that exist in a wide range of shapes and sizes. The variants may differ considerably from a dermatoscopic standpoint. [6705 images]
2. **Melanoma [mel]** is a cancerous tumor that develops from melanocytes and can take many different shapes. If detected early enough, it can be treated with a simple surgical excision. [1113 images]
3. **Benign Keratosis-like Lesions [bkl]** is a flat form of seborrheic keratosis and lichen-planus-like keratoses (LPLK), which is a seborrheic keratosis or solar lentigo with inflammation and regression. [1099 images]
4. **Basal Cell Carcinoma [bcc]** is an epithelial skin cancer that rarely spreads but can be fatal if left untreated. [514 photos]
5. **Vascular lesions [vasc]** range from cherry angiomas to angiokeratomas and pyogenic granulomas and can be benign or malignant.[142 photos]
6. **Actinic Keratoses [akiec**] are a non-invasive kind of squamous cell carcinoma that can be treated locally without surgery. [327 photos].
7. **Vascular lesions [vasc]** range from cherry angiomas to angiokeratomas and pyogenic granulomas and can be benign or malignant. [142 photos]

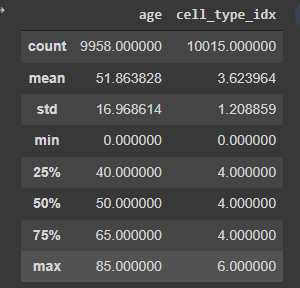
**The HAM10000 dataset:**



**Table 1:** The HAM10000 data set contents.

**The main descriptive statistics of the HAM10000 dataset:**

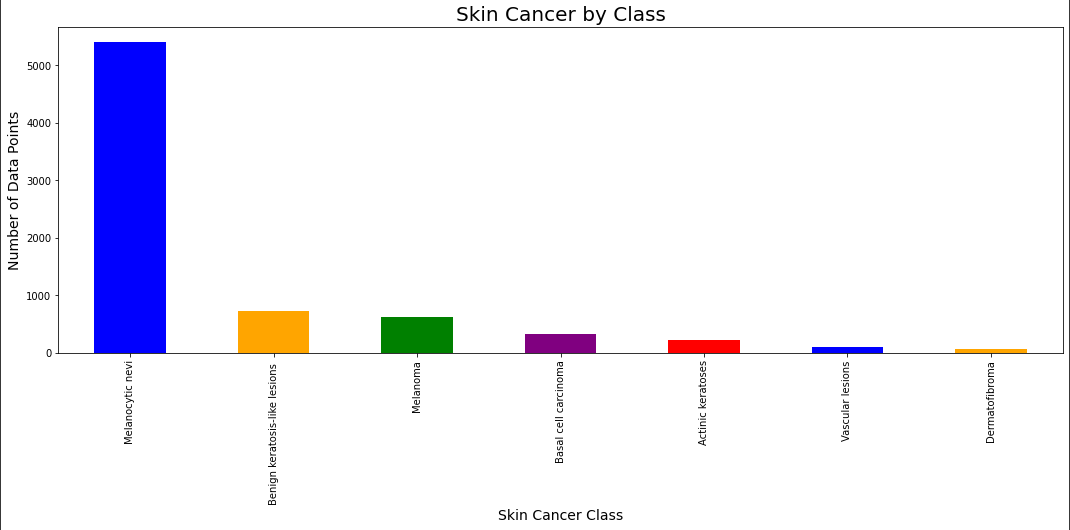


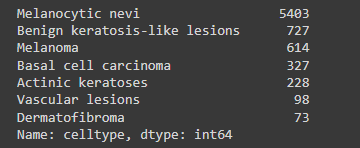
****

**Table 2:** The descriptive statistics data set contents.

We can observe that each entry has a unique image id, but not a unique lesion id, as seen above. This meant that duplicate pictures with the same lesion id but different distortions, such as angle, shear, or zoom distortion, existed. Furthermore, class Melanocytic Nevi [nv] dominated the skin lesion categories, accounting for 6,705 of the 10,015 pictures received, indicating a problem with class imbalance in the data set.

**Graphically Analysis on the HAM10000 dataset:**

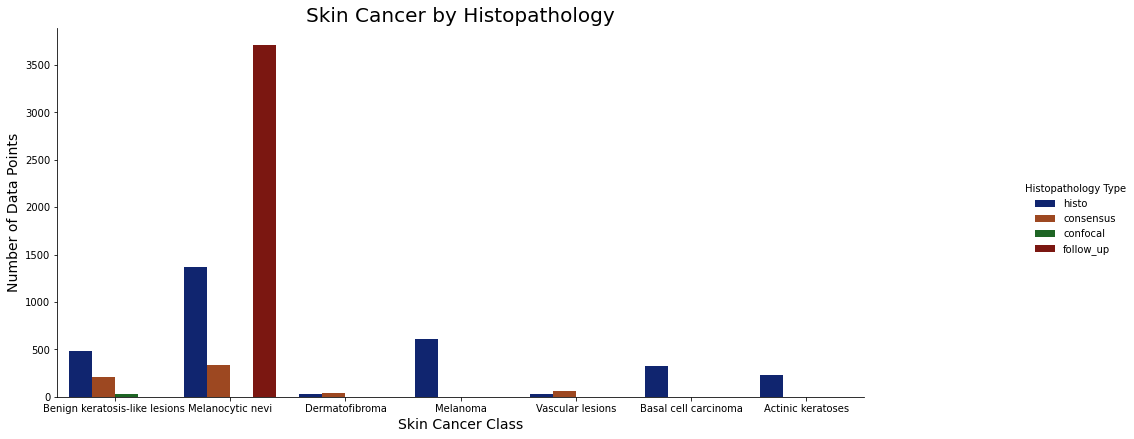
****

****

**Fig 1:** The count of different types of classes

**Analysis of each attributes with target variables:**

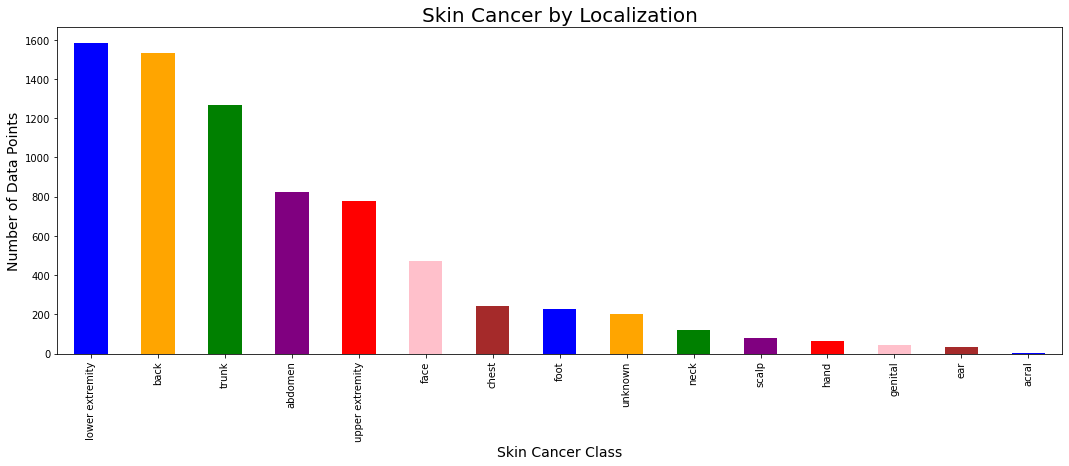
**Skin Cancer by Hispathology:**



**Fig 2:** Skin cancer by Hispathology

We can see from the diagram above that the majority of the technical validation was done using histograms or follow-up. The majority of Melanocytic nevi [nv] confirmation is done by follow-ups.

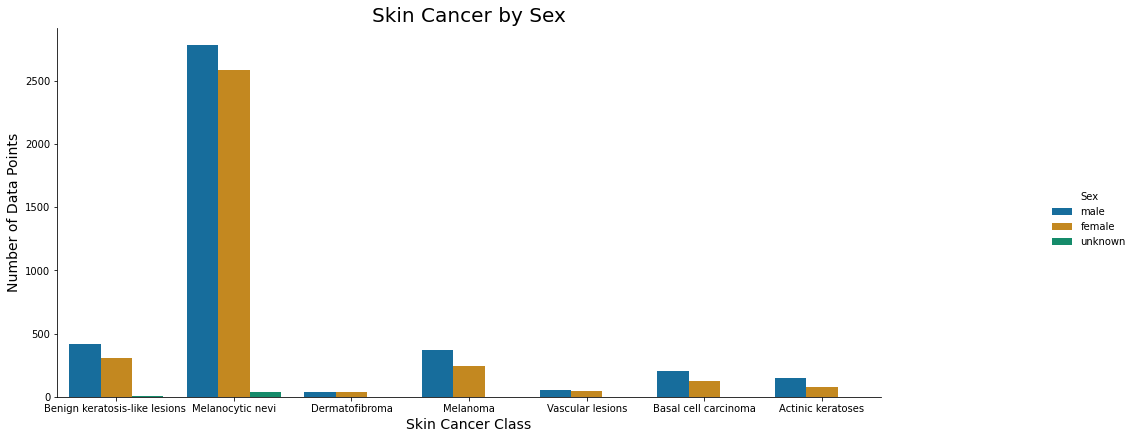
**Skin Cancer by Body Localization:**



**Fig 3:** Skin cancer by Body Localization

We can see that skin cancer as more occurrence in the back, lower extremity and trunk of the people.

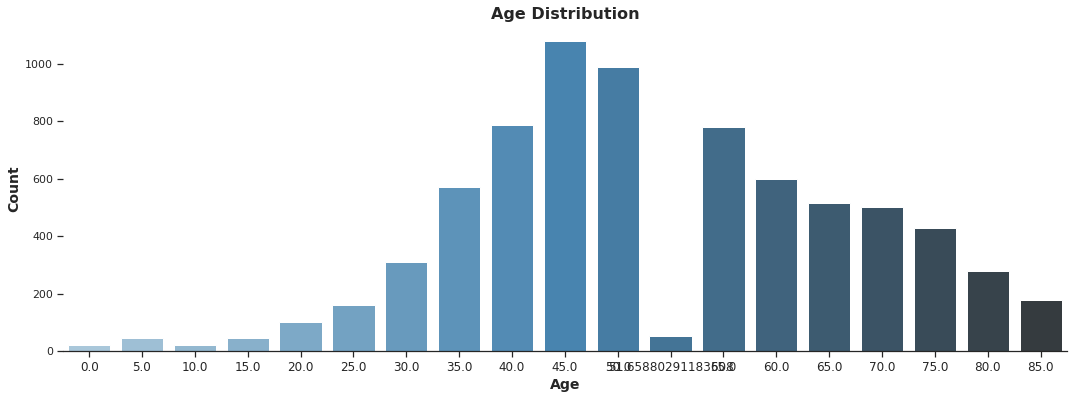
**Skin Cancer by Gender:**



**Fig 4:** Skin cancer by Gender

We can see that cancer kinds are distributed equally across men and women, with Melanocytic nevi [nv] being the most prevalent malignancy in both sexes.

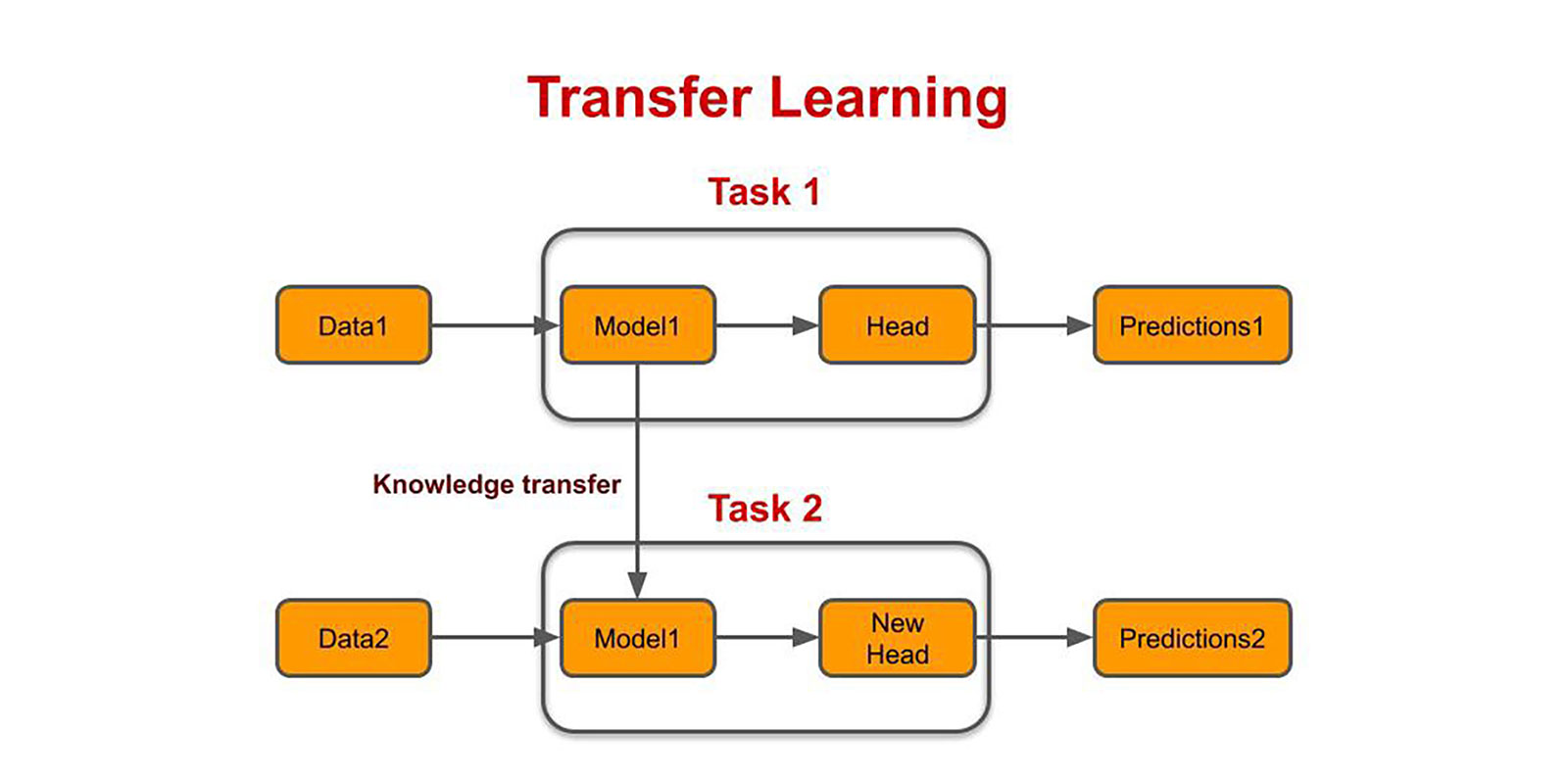
**Skin Cancer by Age:**



**Fig 5:** Skin cancer by Age

Again, we can see from the graph above that Melanocytic nevi [nv] is the most prevalent cancer in the sample; it appears to occur in middle age, whilst other cancer classes appear to develop at a later age.

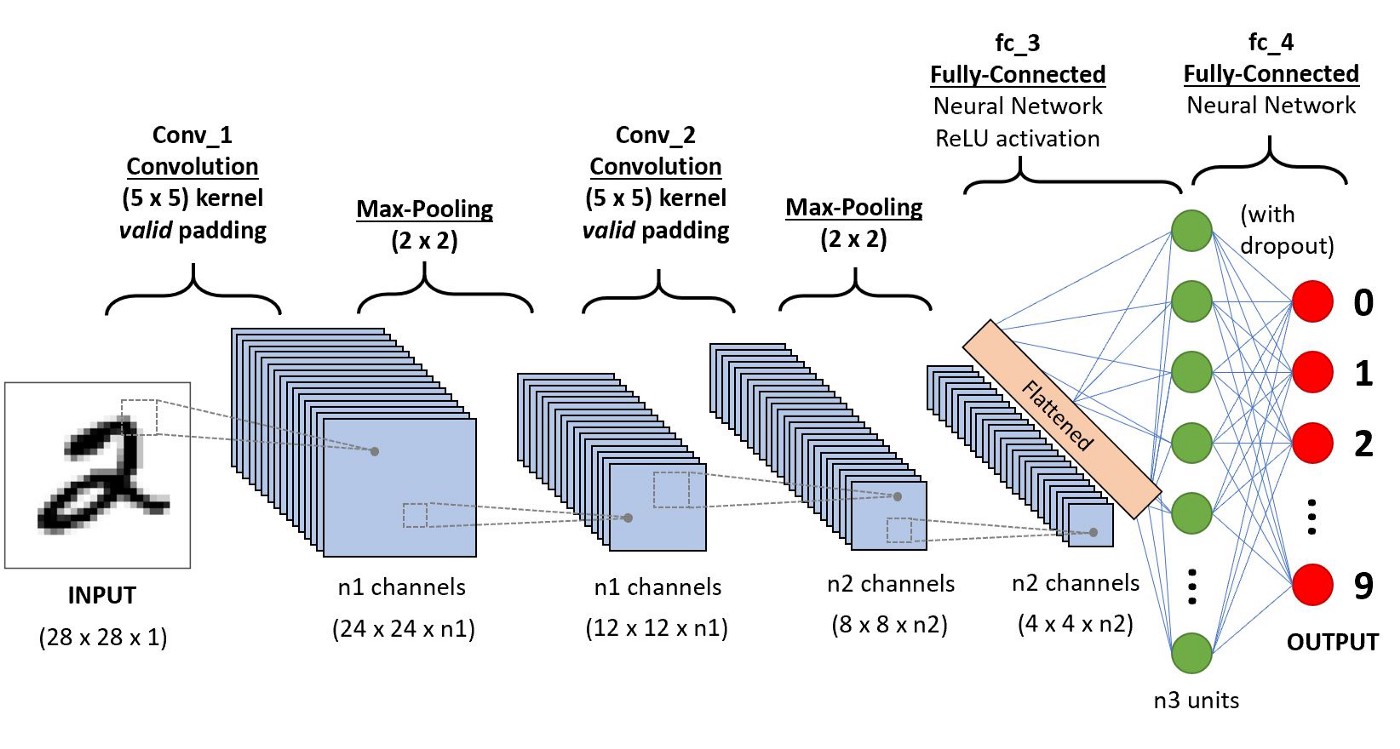
**Transfer Learning:**



**Fig 6:** Flow chart of Transfer Learning

Transfer learning (TL) is a machine learning (ML) research subject that focuses on storing and transferring knowledge learned while addressing one problem to a different but related problem. For instance, skills learned when learning to recognize vehicles can be applied to recognizing trucks. Although practical connections between the two fields are limited, this area of research has some ties to the lengthy history of psychological literature on learning transfer. Reusing or transferring information from previously learned tasks for the learning of new tasks has the potential to boost a reinforcement learning agent's sample efficiency dramatically.

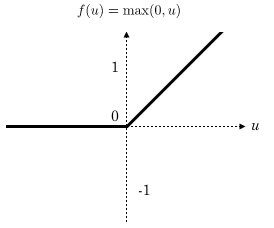
**Convolution Neural Network:**



**Fig 7:** CNN Model

Convolution layers, pooling layers, and fully connected layers are among the building components of the CNN architecture. A typical design comprises of one or more completely linked layers followed by a stack of many convolution layers and a pooling layer. Forward propagation refers to the process of transforming input data into output data using these layers (Fig. 1). Although the convolution and pooling methods discussed in this section are for two-dimensional (2D)-CNNs, comparable operations may be performed on three-dimensional (3D)-CNNs as well.

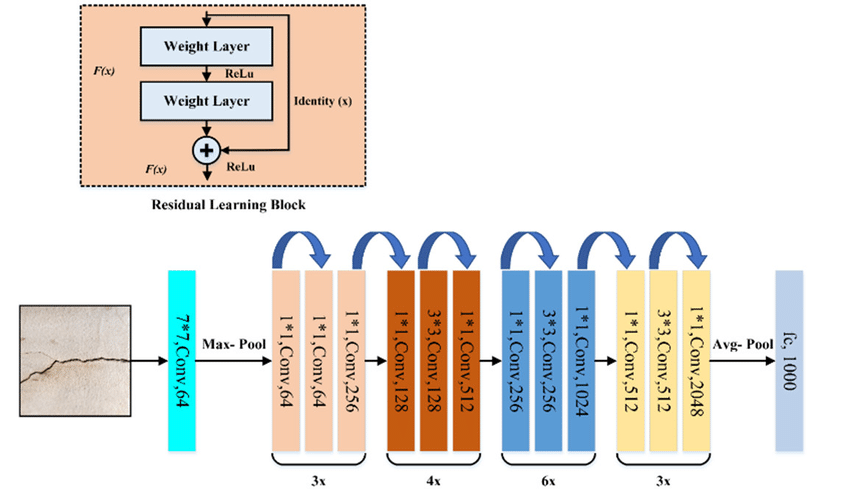
**Layers of Convolution:** The convolutional layer, the pooling layer, the ReLU correction layer, and the fully-connected layer are the four types of layers in a convolutional neural network.

* **Convolution Layer:** Convolutional neural networks' fundamental component is the convolutional layer, which is always at least the first layer. Its goal is to find a set of features in the photos that are given to it as input. Convolution filtering is used to achieve this. As a result, the convolutional layer takes multiple images as input and calculates the convolution of each with each filter. The filters match the features we're looking for in the photographs to a tee. They are started and then updated via gradient descent back propagation.
* **Pooling Layer:** This layer is frequently sandwiched between two convolutional layers: it receives many feature maps and applies the pooling operation to each of them. The pooling procedure reduces the size of the photos while maintaining their essential properties. To accomplish this, we divide the image into regular cells and then keep the highest value in each cell. Small square cells are frequently utilized in practice to avoid losing too much data. The network's pooling layer minimizes the number of parameters and calculations. This increases network efficiency and prevents over-learning.
* **ReLU Correction Layer:** The real non-linear function defined by ReLU(x)=max is referred to as ReLU (Rectified Linear Units) (0,x). In terms of appearance, it appears as follows: 

All negative values received as inputs are replaced by zeros by the ReLU correction layer. It serves as a mode of activation.

* **The fully connected layer:** The fully-connected layer, whether convolutional or not, is usually the last layer of a neural network, therefore it isn't unique to CNNs. This layer takes an input vector and turns it into a new output vector. It accomplishes this by applying a linear combination and, perhaps, an activation function to the incoming input values. The image is classified as an input to the network by the last fully-connected layer, which produces a vector of size N, where N is the number of classes in our image classification issue. Each member of the vector represents the likelihood that the supplied image belongs to a specific class. Weight values are learned in the same way as convolutional neural networks learn convolution layer filters: during the training phase, by back propagation of the gradient.

**ResNet50** Architecture**:**

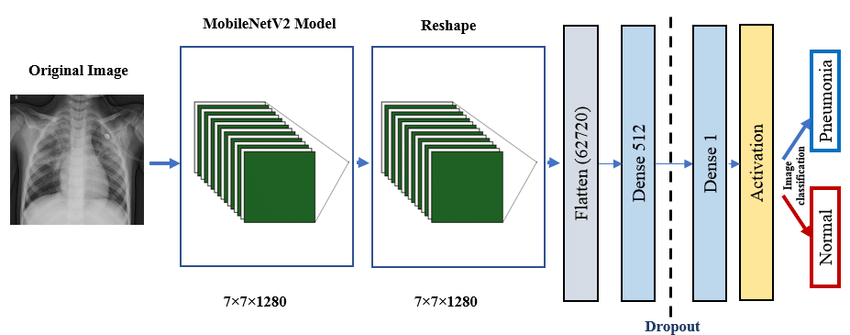


**Fig 8:** ResNet50 Architecture

ResNet50 is a ResNet variation of 48 Convolution layers, 1 MaxPool layer, and 1 Average Pool layer. There are 3.8 x 109 floating point operations in it. It's a popular ResNet model, and we've gone through the ResNet50 design in detail. We'll start with some background information, then compare ResNet50 to other models before diving right into the design. AlexNet took first place in the LSVRC2012 classification challenge in 2012, and ResNet became the most fascinating thing to happen in the computer vision and deep learning worlds after that. Because of the foundation that ResNets provided, it was possible to train ultra-deep neural networks, which means that a network may have hundreds or thousands of layers and still function well. The ResNets framework was initially used to image identification tasks, but as stated in the paper, it can also be utilized for non-computer vision activities to improve accuracy. Many of you may wonder why there was a need for Residual learning for training ultra-deep neural networks when simply stacking more layers provides us improved accuracy.

We used pre-trained ImageNet model weights in the project and then fine-tuned all network layers with our dataset. During the pre-processing procedures, all of the photos are resized. The function extractor is the first layer of the CNN, and the softmax classifier is the final layer, which sorts the images into one of the predefined categories.

**ResNet152V2** Architecture**:**



**Fig 9:** ResNet152V2 Architecture

A CNN architecture with hundreds or thousands of convolutional layers is known as a Residual Network (ResNet). Additional layers' efficacy was reduced by previous CNN configurations. ResNet has a large number of layers and is quite performant. The main difference between ResNetV2 and the original (V1) is that V2 applies batch normalisation to each weight layer before applying it. ResNet performs well in picture recognition and localization tasks, demonstrating the importance of numerous visual recognition applications.

**Data Pre-Processing:**

The data was directly imported into Google Colab from Kaggle. I made two dictionaries for the photos and their labels. The first dictionary contains image names collected from the multiple image folders in the downloaded kaggle data set. The diagnostic skin lesion categories code was then matched to the whole name of the category in a second dictionary.

The process of repairing and eliminating erroneous records from a database or table is known as data cleaning or cleansing. In general, data cleaning or cleansing entails finding and replacing data and records that are missing, erroneous, irrelevant, or otherwise problematic ('dirty').

Scaling and post-processing were done on the photos. Because of the vast amount of photos we had on hand, training our CNN models would have been difficult. To speed up the process and ensure that our CNN models worked properly, images were scaled by a factor of 1:4. The new photos were 150 by 120 pixels in size. The image was flattened and saved as a numeric image list after that.

One-hot encoding is the conversion of categorical data variables into variables that machine learning algorithms can use to improve predictions. One hot encoding is an important aspect of machine learning feature engineering. Allowing the model to presume a natural ordering between categories while using this encoding may result in poor performance or surprising results (predictions halfway between categories). In this scenario, the integer representation can be encoded with a one-hot encoding. For each unique integer value, the integer encoded variable is removed and a new binary variable is added.

**Data Splitting:**

The flattened numeric images list served as the project's feature, while the one-hot encoding for the skin lesion categories served as the project's aim. To guarantee that there were enough samples from each class in each split for appropriate modelling, the data was split 70:10:20 across each class individually.

**Data Augmentation:**

All of the original photos were edited and enhanced at each epoch and then utilised for training to avoid over fitting. The model got more durable and accurate as it was trained on multiple variations of the same image. There were the same number of photos in each period as there were in the original images. The images were as follows:

Randomly rotated by 20%

Randomly shifted horizontally by 20%

Randomly shifted vertically by 20%

Randomly sheared by 10%

Randomly zoomed by 10%

Randomly channel shifted by 10%

**Modelling and Results:**

The ResNet50 and ResNet152V2 models were built ahead of time and loaded using the keras application library package. To align parameter numbers and outputs, extra layers at the top (final layers) were added.

**Models hyper parameters:**

Hyper parameter - Value

Optimizer - Adam

Loss Function Categorical Cross - Entropy

Epochs - 25

Batch Size – 10

**Optimizer:** Adam is the most extensively used optimization technique for deep neural network training today because it is simple to use, computationally efficient, and effective when dealing with large volumes of data and parameters.

**Loss Function**: A loss function for categorizing single labels is categorical cross-entropy. When just one category applies to each data point, this is the case. This worked well in this situation because each of the seven types of skin lesions could only be represented by one sample.

**Epochs:** After a series of tests with values of 20, 25, 50, 100, 150, and 200 epochs, it was discovered that 50 epochs was sufficient for the best outcomes.

**Batch Size:** A batch size of 10 produced the greatest results after a series of trials with batch sizes of 5, 10, 20, and 40.

**Evaluation Metrics:**

There are a lot of metrics that can be used to evaluate a classification task, but we'll focus on accuracy, precision, and recall.

• **Accuracy:** A model's accuracy is measured by the proportion of correct forecasts to total predictions. The more precise the model is, the more accurate the class forecasts are.

**• Recall:** A model's recall is defined as the ratio of true positives to total true positives and false negatives.

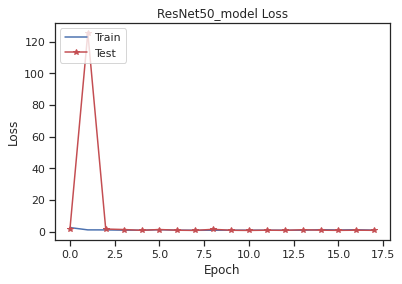
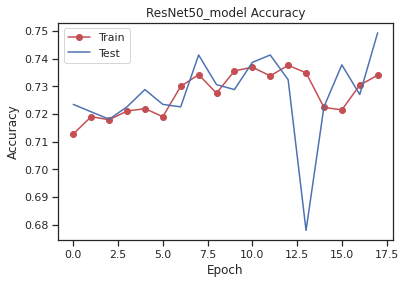
• **Precision:** The proportion of true positives to the total number of true positives and false positives is the precision of the model. The lower the false positive predictions the model generates for that class, the higher the precision.

**RESULTS:**

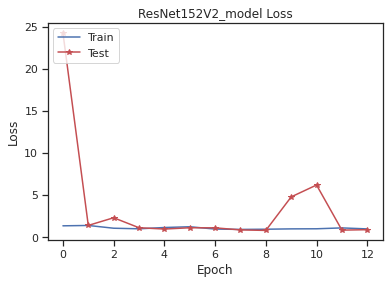
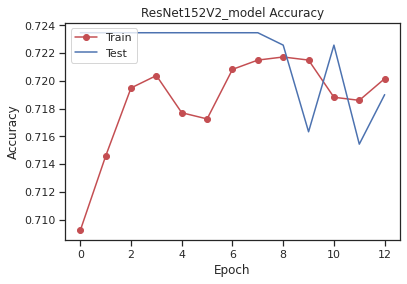
**ResNet50:** Default ResNet50 The ResNet50 model requires an image size of (224,244,3), but our image sizes differ, thus we must adjust the image size parameter in the first layer. The size of our group will be (150,120,3). We are omitting the top layer and altering it because our problem has seven classes.

**ResNet152V2:** Similarly, the ResNet152V2 model expects the size to be (200,200,3), thus we'll need to update the image size parameter in the first layer as before.

The accuracy and loss gained during the training and testing procedure for ResNet50 and ResNet152V2 are plotted on a graph.



**Fig 10:** Plots of loss and accuracy for ResNet50 model.



**Fig 11:** Plots of loss and accuracy for ResNet152V2 model.

It was determined that after 25 epochs of training, the model had a test accuracy of 73 percent for both the model and the data (ResNet50 and ResNet152V2). The accuracy of training data for the ResNet50 model increased in the first epoch, declined in the second epoch, and remained constant in the third epoch throughout the procedure. ResNet50's test accuracy was nearly identical across all epochs. ResNet50's training and test loss grew for the first four epochs, then gradually dropped to two epochs. After that, it began to rise once more.

Similarly, the training and test accuracy of ResNet152V2 increased steadily until the eighth epoch, when it plummeted to its lowest level. After the first epochs, ResNet152V2's training and test loss dropped, then remained consistent throughout the procedure.

**PERFORMANCE RESULTS OF TRANSFER LEARNING:**

|  |  |  |  |
| --- | --- | --- | --- |
| **CNN models** | **Accuracy** | **Precision** | **Recall** |
| ResNet50 | 0.723769 | 0.628929 | 0.723769 |
| ResNet152V2 | 0.722163 | 0.522852 | 0.722163 |

**Conclusion:**

We used two alternative CNN architectures (ResNet50 and ResNet152V2) to predict skin lesion types based on skin lesion photographs in this project. Both models had the same accuracy of 0.72, precision of 0.6, and recall of 0.73.While both models correctly predicted Melanoma when tested on random photos, they occasionally failed to predict other skin lesions. Two of the most widely used CNN models for image categorization were successfully built and used (ResNet50 and ResNet152V2).

There is a tiny chance of boosting the accuracy and precision by having a few more additional attributes and model upgradation by parameters, which can be done further. As a result, detections can be more specific and accurate, resulting in improved therapy.