SKIN CANCER DETECTION USING CONVOLUTIONAL NEURAL NETWORK

Under the Guidance of

Dr. Agilandeeswari L

(Associate Professor, School of Information Technology and Engineering)

For the Course

ITE1015 – Soft Computing

For Winter Semester 2020-21

Project Team:

Aashish Bansal 19BIT0346

Perumalla Sasank 19BIT0338

Keerthi Yasasvi 19BIT0335

TABLE OF CONTENTS

SKIN CANCER DETECTION – GENERAL RESEARCH

GENERAL RESEARCH STUDY

ABSTRACT

The project is a CNN trained model which can predict whether the patient has a suffering from Cancer or not by checking the images of the infected areas on the body. The model has been trained on a variety of images through which it predicts the required.
In this project, the image file of the patient is upload into a software, which is GUI-based interface, developed with the help of Tkinter, and it consists of the model saved as a file and the software uses that to analyze the image and give the prediction which can help doctors to start with the medication way faster instead of waiting for the laboratory reports for the confirmation.
So basically,

- Skin cancer is an abnormal growth of skin cells. Most skin cancers are caused by exposure to ultraviolet (UV) light. When the skin is not protected, UV rays from sunlight or tanning beds can damage and alter skin's DNA that leads to the cancer.
- Deep learning model has been built to classify and identify the binary diagnostic group of melanocytic images obtained through dermoscopy.
- Based on the model, disease detection through dermal cell images has been investigated, and classifications on dermal cell images have been performed.

KETWORDS

- Model
- Convolutional Neural Network
- Cancer
- Malignant
- Benign
- Detection
- Tkinter
- Software
- Analysis
- Uploading
- Training
- Testing
- Validation
- Prediction
- MobileNet
- Inception
- Xception

INTRODUCTION

Skin cancer is the most common form of cancer in the United States. The two most common types of skin cancer—basal cell and squamous cell carcinomas—are highly curable, but can be disfiguring and costly to treat. Melanoma, the third most common skin cancer, is more dangerous and causes the most deaths. The majority of cases of these three types of skin cancer are caused by overexposure to ultraviolet (UV) light.

The most common warning sign of skin cancer is a change on the skin, typically a new mole, a new skin lesion or a change in an existing mole.

- Basal cell carcinoma may appear as a small, smooth, pearly, or waxy bump on the face, or neck, or as a flat, pink/red- or brown-coloured lesion on the trunk, arms or legs.
- Squamous cell carcinoma can appear as a firm, red nodule, or as a rough, scaly, flat lesion that may itch, bleed and become crusty. Both basal cell and squamous cell cancers mainly occur on areas of the skin frequently exposed to the sun, but can occur anywhere.
- Melanoma usually appears as a pigmented patch or bump. It may resemble a normal mole, but usually has a more irregular appearance.

DATASET - **ISIC2017: SKIN LESION ANALYSIS**

ABSTRACT OF DATASET

The goal of the challenge is to help participants develop image analysis tools to enable the automated diagnosis of melanoma from dermoscopic images. Image analysis of skin lesions is composed of 3 parts:

- Part 1: Lesion Segmentation
- Part 2: Detection and Localization of Visual Dermoscopic Features/Patterns
- Part 3: Disease Classification

This challenge provides training data (150 images) and blind held-out test dataset (~600 images) will be provided for participants to generate and submit automated results.

BACKGROUND

MELANOMA

Skin cancer is a major public health problem, with over 5 million newly diagnosed cases in the United States each year. Melanoma is the deadliest form of skin cancer, responsible for over 9,000 deaths each year.

DERMOSCOPY

As pigmented lesions occurring on the surface of the skin, melanoma is amenable to early detection by expert visual inspection. It is also amenable to automated detection with image analysis. Given the widespread availability of high-resolution cameras, algorithms that can improve our ability to screen and detect troublesome lesions can be of great value. As a result, many centres have begun their own research efforts on automated analysis. However, a centralized, coordinated, and comparative effort across institutions has yet to be implemented.

Dermoscopy is an imaging technique that eliminates the surface reflection of skin. By removing surface reflection, visualization of deeper levels of skin is enhanced. Prior research has shown that when used by expert dermatologists, dermoscopy provides improved diagnostic accuracy, in comparison to standard photography. As inexpensive consumer dermatoscope attachments for smart phones are beginning to reach the market, the opportunity for automated dermoscopic assessment algorithms to positively influence patient care increases.

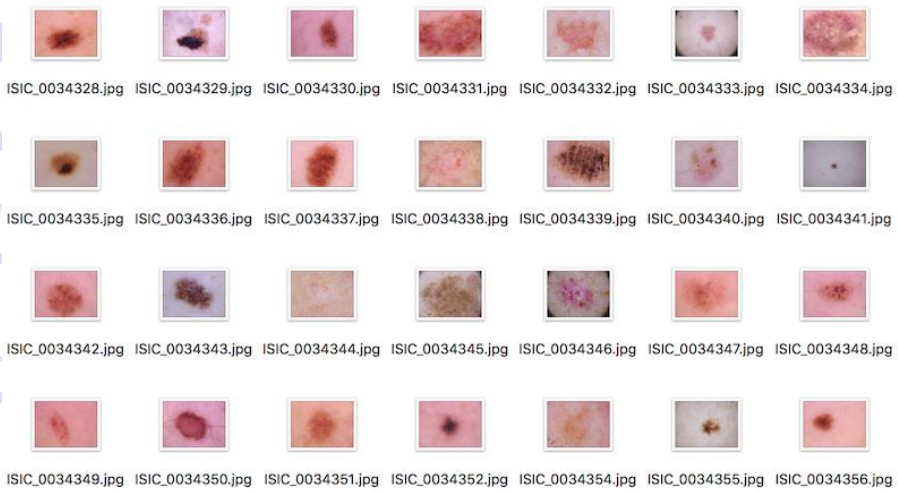

Figure 1: Sample Dataset Images

ABOUT THE DISEASES

Skin cancer is the most prevalent type of cancer. Melanoma, specifically, is responsible for 75% of skin cancer deaths, despite being the least common skin cancer. The American Cancer Society estimates over 100,000 new melanoma cases will be diagnosed in 2020. It's also expected that almost 7,000 people will die from the disease. As with other cancers, early and accurate detection—potentially aided by data science—can make treatment more effective.

Currently, dermatologists evaluate every one of a patient's moles to identify outlier lesions or “ugly ducklings” that are most likely to be melanoma. Existing AI approaches have not adequately considered this clinical frame of reference. Dermatologists could enhance their diagnostic accuracy if detection algorithms take into account “contextual” images within the same patient to determine which images represent a melanoma. If successful, classifiers would be more accurate and could better support dermatological clinic work.

As the leading healthcare organization for informatics in medical imaging, the Society for Imaging Informatics in Medicine (SIIM)'s mission is to advance medical imaging informatics through education, research, and innovation in a multi-disciplinary community. SIIM is joined by the International Skin Imaging Collaboration (ISIC), an international effort to improve melanoma diagnosis. The ISIC Archive contains the largest publicly available collection of quality-controlled dermoscopic images of skin lesions.

In this competition, you’ll identify melanoma in images of skin lesions. In particular, you’ll use images within the same patient and determine which are likely to represent a melanoma. Using patient-level contextual information may help the development of image analysis tools, which could better support clinical dermatologists.

Melanoma is a deadly disease, but if caught early, most melanomas can be cured with minor surgery. Image analysis tools that automate the diagnosis of melanoma will improve dermatologists' diagnostic accuracy. Better detection of melanoma has the opportunity to positively impact millions of people.

MOTIVATION

- Disease detection plays a very important role in the process of diagnosis. Therefore, the motivation lies in accurate classification and detection of the diseases based on medical images.
- The main aim is to minimize the chances of error that might happen due to the doctor's misjudgement.
- Developing a system that will not only help in detecting the diseases efficiently but will also save the time and effort of the medical practitioners.
- This will also save the patients from running to the doctor to get their medical reports verified.

GENERAL ARCHITECTURE

(Analytics Vidya, 2019)

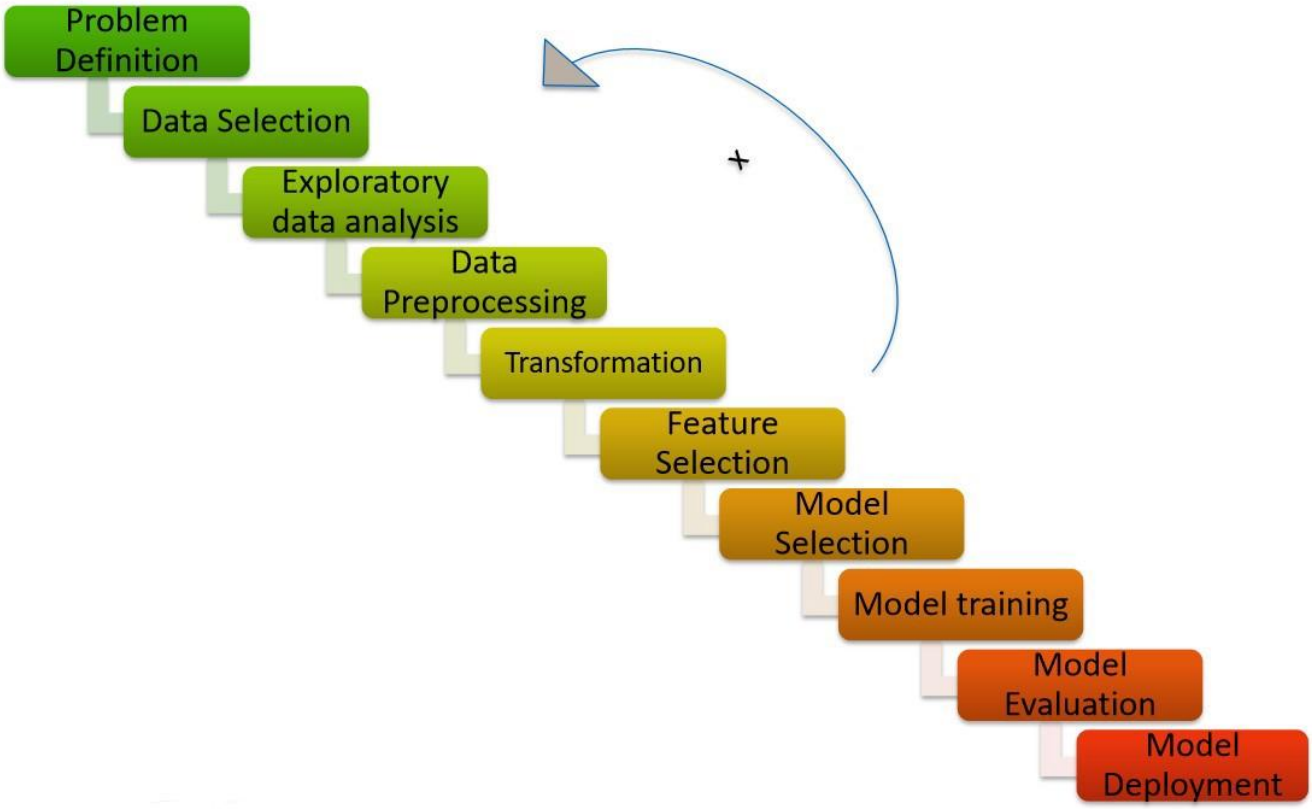

Figure 2: General Architecture

SUMMARY ON THE GENERAL ARHITECTURE PROCESSES

DATA SELECTION

The data selection is done on the basis of the amount of data and the type of data which is available. The data could be in the form of images, scanned reports, tf records, dcm files, etc. Based on the algorithm selected and the kind of data available, the model will be built. The data obtained can be collected via survey or from public databases.

EXPLORATORY DATA ANALYSIS

CHECKING THE TYPES OF DATA

To find what all columns it contains, of what types and if they contain any value in it or not, with the help of functions.

FINDING THE OUTLIERS

An outlier is a piece of data that is an abnormal distance from the other points. In other words, it's data that lies outside the other values in the set. These points can be found by plotting the entire data.

DATA VISUALIZATION

Using this data, we can:

- Analyse individual columns
- Check for missing values
- Perform variable analysis
- Check condition column
- Check quality column
- Plot between different variables and targets

DATA PRE-PROCESSING

SPLITTING THE DATA

It is very important because your model needs to be evaluated before it has been deployed. And that evaluation needs to be done on unseen data because when it is deployed, all incoming data is unseen. The main idea behind the train test split is to convert the original data into training and testing data. For most of the articles which have been analysed, the data has been split into training and testing data in the range of ration of 75% to 25% (this is an approximate range provided considering all the research papers analysed), respectively.

CHECKING FOR MISSING VALUES

If your data set is full of NaNs and garbage values, then surely your model will perform on garbage too. So, taking care of such values is important and it mostly done using the Simple Imputer method.

CHECKING CATEGORICAL FEATURES

The most common methods used for this are:

- Label Encoding
- One-hot Encoding

NORMALIZING DATASET

The models mostly use the following methods of normalization for the data:

- Standard Scaler
- Variance before Standard Scaler
- Variance after Standard Scaler

FEATURE TRANSFORMATION

Feature pre-processing is one of the most crucial steps in building a Machine learning model. Too few features and your model won't have much to learn from. Too many features and we might be feeding unnecessary information to the model. Not only this, but the values in each of the features need to be considered as well.

WHY DO WE NEED FEATURE TRANSFORMATION AND SCALING?

Oftentimes, we have datasets in which different columns have different units – like one column can be in kilograms, while another column can be in centimetres. Furthermore, we can have columns like income which can range from 20,000 to 100,000, and even more; while an age column which can range from 0 to 100(at the most). Thus, Income is about 1,000 times larger than age.

But how can we be sure that the model treats both these variables equally? When we feed these features to the model as is, there is every chance that the income will influence the result more due to its larger value. But this doesn't necessarily mean it is more important as a predictor. So, to give importance to both Age, and Income, we need feature scaling.

FEATURE TRANSFORMATIONS USED IN THE MODELS

MAXABS SCALAR

In simplest terms, the MaxAbs scaler takes the absolute maximum value of each column and divides each value in the column by the maximum value.

Thus, it first takes the absolute value of each value in the column and then takes the maximum value out of those. This operation scales the data between the range [-1, 1].

ROBUST SCALAR

If you have noticed in the scalers we used so far, each of them was using values like the mean, maximum and minimum values of the columns. All these values are sensitive to outliers. If there are too many outliers in the data, they will influence the mean and the max value or the min value. Thus, even if we scale this data using the above methods, we cannot guarantee a balanced data with a normal distribution.

The Robust Scaler, as the name suggests is not sensitive to outliers. This scaler-

removes the median from the data

scales the data by the Interquartile Range (IQR)

Are you familiar with the Inter-Quartile Range? It is nothing but the difference between the first and third quartile of the variable.

UNIT VECTOR SCALER

Normalization is the process of scaling individual samples to have unit norm. The most interesting part is that unlike the other scalers which work on the individual column values, the Normalizer works on the rows! Each row of the data frame with at least one non-zero component is rescaled independently of other samples so that its norm (l1, l2, or inf) equals one.

Just like MinMax Scaler, the Normalizer also converts the values between 0 and 1, and between -1 to 1 when there are negative values in our data.

However, there is a difference in the way it does so.

- If we are using L1 norm, the values in each column are converted so that the sum of their absolute values along the row = 1
- If we are using L2 norm, the values in each column are first squared and added so that the sum of their absolute values along the row = 1

FEATURE SELECTION

Feature selection is a process where you automatically select those features in your data that contribute most to the prediction variable or output in which you are interested.

Having irrelevant features in your data can decrease the accuracy of many models, especially linear algorithms like linear and logistic regression.

Three benefits of performing feature selection before modelling your data are:

- **Reduces Overfitting**: Less redundant data means less opportunity to make decisions based on noise.
- **Improves Accuracy**: Less misleading data means modelling accuracy improves.
- **Reduces Training Time**: Less data means that algorithms train faster.

The methods used for Feature Selection are:

- Principal Component Analysis
- Linear Discriminant Analysis

PRINCIPAL COMPONENT ANALYSIS

Principal Components Analysis is a way of recognizing patterns in data, and expressing the data in such a manner as to focus their differences and similarities. Subsequently patterns in data may be complex to discover in data of high dimension, where the luxury of graphical representation is not available, PCA is a powerful tool for analysing data. The key advantage of PCA is that once we have found the patterns in the data, and you compress the data, i.e., by reducing the number of dimensions, without much loss of information. This technique used in image compression.

LINEAR DISCRIMINANT ANALYSIS

There are many possible techniques for classification of data. Principal Component Analysis and Linear Discriminant Analysis are commonly used techniques for dimensionality reduction and data classification. Linear Discriminant Analysis easily handles the case where the within-class frequencies are unequal and their performances have been examined on randomly generated test data. This technique maximizes the proportion of between-class variance to the within-class variance in any specific data set in that way promising maximal separability. The Linear Discriminant Analysis is used for classification issues such as speech recognition. The key difference among LDA and PCA is that PCA perform feature classification and LDA works for data classification. The shape and location of the inventive data sets changes when transformed to a different space in PCA, on the other hand LDA doesn't change the location but only attempts to offer more class separability and induce a decision region among the given classes. This technique also supports to better recognize the distribution of the feature data.

MODEL SELECTION

Some of the models which were deployed are:

- Simple Convolutional Neural Network Models
- Transfer Learning Models
- Ensemble Models
- Simple K-Means Model
- Generative Automotive Networks

MODEL TRAINING

The models deployed one of the following training techniques:

- Infected Area Detection
- Image Classification
- Instance Segmentation

MODEL EVALUATION

Some of the most common evaluation methods are:

- Accuracy
- Sensitivity
- Specificity
- Recall
- Precision
- F-measure

Some of the rare evaluation methods used are:

- CNN with/without Data Augmentation
- ResNeXt WSL
- ABCD Rule
- GOPS
- L1D Miss Rate
- XGBoost
- GoogLeNet/ResNet/AlexNet/VGGNet Error Rate

COMPARATIVE STUDY ON VARIOUS SUBTITLES:

LITERATURE SURVEY

The Literature Survey done for the accomplishment of the project is:

Authors &Year	Methodology or Techniques used	Advantages	Issues	Metrics used	Pros	Cons
May-20	CNN, AlexNet, ResNet-18, VGG16, SVM, Black-hat filter, Inpaint Algorithm, Median Filter, Otsu's Methodology	SVM Accuracy = 86.21%, ResNet Accuracy = 87%	Accuracy Original Data = 80%, Accuracy Augmented Data = 98.61%	ReLU, CNN with Data Augmentation = 88.87%, CNN without Data Augmentation = 78.96%	Good Accuracy	Small Dataset
2020	CNN, Inception-v3, Keras, TensorFlow, DCNN, Leaky ReLU, Adamax optimizer, TPR	0.86 AUROC for BKL	0.78 AUROC for MEL	Accuracy	N/A	N/A

	is similar to the positive predictive value					
2020	MVSM classifier, CNN, feature extraction, GLCM, SVM, ABCD	dataset which consists of eight different classes is compressed into 800 images and applied, the accuracy achieved is about 96.25%.	accuracy is lowered if minute amounts of foreign elements are found on the sample	Accuracy	Eight Classes help specify the disease for specific medication	High accuracy on a very small specific training dataset
2020	CNN SENet154, WSL, Adam, weighted loss-entropy	Efficient Architecture	Not much improved with ensemble strategy	EfficientNet, SENet (T1 = 67.2%, T2 = 70.0%), ResNeXt WSL (T1 = 65.9%, T2 = 68.1%)	Transfer Learning	Not implemented properly with small dataset, parameter tuning required
2020	GLCM, HOG, GAC	Feature extraction for early detection	Not enough/adequate dataset	ABCD Rule, SVM Classifier, Accuracy, Sensitivity, Specificity using KNN	characterize the texture of an image by calculating how often pairs of pixels with spe cific values and in a specified spatial relationship occur in an image	high dimensionality of the matrix and the high correlation of the features
2019	Multiclass SVM, AlexNet, ReLU	Accuracy – 94.016%	Model used is a pre-trained model, robust	GOPS, L1D miss rate	Excellent feature extraction	There are many scenarios in which there are multiple categories to which points

						belong, but a given point can belong to multiple categories. In its most basic form, this problem decomposes trivially into a set of unlinked binary problems
2019	CNN, pooling layers, dense network, SVM	AlexNet, VGG16, ResNet-18	Deep Network but Small Dataset – 3000 images	Accuracy = 74%	N/A	The model does not predict properly.
Apr-19	CNN, pooling layer, dense network	Accuracy – 89.5%	Time consuming	Accuracy = 89.5%, Recall = 0.84, Specificity, Precision = 0.8325, F-measure = 0.8325	Images have been taken randomly for better training and for wider input category of features.	The model may not predict that well on the given but may be able to identify the features more accurately.
Mar-19	CNN, Inception V2 Net, K-means Cluster, Max-pooling, Sonification Algorithms	No. of K-means Epochs = 100	F2-score +ve Prediction = 59.9%, High Sensitivity, Low Specificity	F2-score = 81.8%, Sensitivity = 91.7%, Specificity = 41.8%, Precision = 57.3%	Technology improves accuracy of skin cancer diagnosis and might assist physicians to diagnose skin cancer and bypass dermoscopy-related experience factors, time constraints, physical inconvenience of acquiring images	no a priori technology which can identify whether a suspicious lesion is mild, moderate or severely dysplastic

2019	CNN, McNemar Test, ResNet50, Bonferroni Correction	MATLAB	Small dataset (11,444), training may be inefficient, class imbalance	Accuracy = 100%, detection rate = 100%	The prediction may be correct most of the time.	The model might have overfit on the data so it might predict properly on a specific kind of data which might be fed to it.
2018-2019	CNN, VGG16, ImageNet	Accuracy – 92.5%	F1-score = 0.77, VGG16 Accuracy = 78%	Random Forest = 65.9%, XGBoost = 65.15%, SVM = 65.86%, ReLU, Sigmoid	Max-pooling fetches maximum pixel	Low F-score
2018	MatConvNet & GoogLeNet Inception V3 CNN, GoogLeNet, AlexNet, ResNet, VGGNet, Simple Majority Voting, SMP	1.28 million NATURAL images 500 epochs, pre-trained models used, MatConvNet provides pre-trained CNN models and some functions to create and initialize new neural networks	Limited computational resources, time-consuming procedures	GoogLeNet Error Rate = 0.1, ResNet Error Rate= 0.02, AlexNet Error Rate = approx. 0.001, VGGNet Error Rate = approx. 0.001	High Results for all the evaluation metrics	Too many evaluation metrics and parameters used
2018	CNN, ImageNet, AlexNet, VGG, GoogLeNet, ResNet	Big dataset	There is a risk of overfitting the neural network	Accuracy, Image classification	advantageous for the decision making of dermatologists	There is a risk of overfitting the neural network

2020	skin lesions using CNNs, Alexnet. deep learning with PNASNet-5	large dataset with 4 classifications	Less accuracy	Good dataset	efficient in computing time, consume less memory	N/A
2018	CNN, AlexNet, deep learning, AlexNet, VGG, GoogLeNet, ResNet, Inception V3	Public dataset and ISIB-2016,2017	Time-consuming, and errors	Large variety, dataset	advantageous for the decision making of dermatologists	Time-consuming. In addition, errors and the loss of information in the first processing steps have a very strong influence on the classification quality
2020	deep convolutional neural network, computer image analysis algorithms, CNN , GoogLeNet Inception V3, AlexNet Deep Learning CNN	More variants from ISIC	Accuracy of less than 75%	Accuracy but small dataset	metric area under the curve of 99.77% was observed.	This would consume time and the patient may advance to later stage
2019	Machine learning,	Faster identification	Less accuracy	Accuracy, time	Got accuracy of min 85%	highly complex and expensive diagnosis with difficulties and subjectivity of human interpretation
2017	deep neural networks, Deep convolutional neural	dataset of 129,450 clinical images of	less variants	CNN; melanoma;	Large dataset and accuracy	Less variants of Malignant and benign

	networks (CNNs)	Malignant and benign		skin cancer; image pre-processing		
2020	deep learning, CNN, AlexNet and VGG-16, including VGG-Net, ResNet50, InceptionV3, Xception, and DenseNet121	70 % images were used for training and 30% used for testing	Less accuracy	Good dataset	70 % images were used for training and 30% used for testing	accuracy of 65% to 75%, time consuming
2017	SVM, CNN, MobileNet	High accuracy	Small dataset	High accuracy	High accuracies in most cases	High Cost of pre trained models which are required
2018	CNN, GLCM, deep learning, CNN, ResNet, InceptionV2	Trained on many variants	Small dataset	Accuracy but small dataset	Accuracy increases with bigger dataset	Bigger dataset required, Time consuming process
2020	CNN, SVM, KNN, Naïve Bayes, and neural network	97.8 % of Accuracy	High rate of overfitting and misidentificatio n	accuracy	obtained is 97.8 % of Accuracy and 0.94 Area under Curve using SVM classifiers and additionally the Sensitivity obtained was 86.2 % and Specificity obtained was 85 % using KNN.	High rate of overfitting and misidentificatio n
2017	GANs, CNN, AlexNet, StyleGANs, InceptionV3-	size of 600×600 as input dataset,	sets the weight coefficient w in	Accuracy	Model Automatically learns the feature	Proposed DCGANs, which have clear structural

	StyleGANs, ResNet50-StyleGANs, VGG16BN-StyleGANs		the SoftMax loss function		representations required for the corresponding detection or classification tasks through the dataset, and has a good performance in many applications	constraints and indicate that they have weak credibility for unsupervised learning and that they are generalized most of the time.
2018	CNN, SciKit, Keras, TensorFlow, OpenCV, ReLU	90% accuracy, Convolution maintains the spatial interrelation of the pixels, values of the pixels ranging from 0 - 255 i.e., 256 pixels.	Rectified Linear Unit is a non-linear operation. ReLU acts on an elementary level.	Accuracy	With a large dataset accuracy can be increased to 90%	Averages and accuracy of 70% on standard publicly available dataset and time consuming
2019	AlexNet, Ordinary CNN, VGG-16, LIN, Inception-v3, and ResNet. Lévy flight, ReLU	size of input images in the input is considered 28×28 pixel.	doesn't give the best global solution	Accuracy	97% accuracy	Imbalance of training and testing dataset
2019	STM32, ROC, CNN, ReLU, NLSC	Accuracy - 99%, F1-Score - 99%	computing and index loss, poor lesion skin discrimination specificity	Accuracy	This methodology, based on "visual" investigation by the dermatologist and/or oncologist, has the advantage of not being	Several approaches proposed in scientific literature increase sensitivity of the pipelines to the disadvantage of 'specificity' or vice versa.

					invasive and quite easy to perform	
2019	CNN, Feature Extraction, HSV format	Accuracy of 98%. for melanoma skin cancer detection and 93% for melanoma type, TPR of 94.25%, FPR of 3.56%, and EP of 4%, average accuracy of 91.66%	high error rates, 25.6% Caucasian error and 23.2 Xanthous error, validation loss of 57.56%	Accuracy	achieved TPR of 94.25%, FPR of 3.56%, and EP of 4%	With a small dataset an accuracy of 74.76% and validation loss of 57.56% is acquired
2019	CNN, keras, AlexNet, VGG16, SGD optimiser,	trained on more than 126k images, higher image augmentation (24x) and image resolution (1k), the same performances can be achieved using less than 5000 images, no impact of image resize filters	Experiments at 277x277 pixel resolution, Experiments without transfer learning	Accuracy	98% specificity	73% sensitivity and Jaccard Index of 0.69.
2019	CNN, grad-CAM, TensorFlow, Inception-ResNet-v2, DenseNet121, Xception	consists of 150,223 clinical images from 543 different skin diseases, achieved an accuracy of 87.25 ± 2.24% on the	highest average precision (77.0%)	Accuracy	achieved 92.9%, 89.2%, and 84.3% recalls for the LE, BCC, and SK, respectively,	mean recall and precision reached 77.0% and 70.8%.

		dermoscopic images for four common skin diseases, including SK, BCC, psoriasis and melanocytic nevus.				
2019	CNN, keras, TensorFlow, Inception V3, ResNet50, VGG16, MobileNet and Inception, Resnet	7 types of skin lesion diseases identification namely: Benign Keratosis, Dermatofibroma, Vascular Lesion, Melanoma, Melanocytic Nevus, Basal Cell Carcinoma and Actinic Keratosis., Inception, Resnet achieved an average accuracy of 91%, Accuracies of 90 and 91%	low F1 score	Accuracy	This model is advantageous over feed-forward neural networks which cannot understand translation invariance	Low F1 Scores

GROUPING

ON THE BASIS OF THE DATA USED

The groups for this which can be formed are:

- Dataset is classified into classes and stored separately beforehand
    - Data is classified as:
        - Malignant or Benign

  - 7 classes of Skin Cancer
  - 3 classes of keratosis
- Dataset is not classified into classes and instead CSV file is provided

ON THE BASIS OF THE DATA ANALYSIS TECHNIQUE

Some the analysis methods used are:

- ABCDE Rule
- Data Augmentation
- Normal scanning and cropping and photoshop
- CNN (common)
- Deep Learning Pipeline (Morphological Analysis) (common)
- Biopsy, Histopathological Testing, Dermoscopic Assessment
- StyleGANS (common)
- GLCM, SVM

ON THE BASIS OF DATA PRE-PROCESSING TECHNIQUE

Some of the pre-processing methods can be grouped as:

DULL RAZOR METHOD (COMMON)

Dull Razor software [10] is a medical imaging software for hair removal. In this, a special type of filter is used, which replaces hair pixels by neighboring pixels. It improves classification results Hair removal is done using Dull Razor software. The dermoscopic images may contain hairs. These hairs somehow will give erroneous classification. So, it is desirable to do the hair removal before proceeding to further steps.

PROS

- One of the main advantages of the simulation that the ground truth pixel values and hair mask are known

(a) Dermoscopic image containing hairs (b) Hairs removed using Dull Razor

Fig. 3: Hair removal using Dull Razor Software

Figure 3: Sample Image for Example

Dull Razor algorithm

Step 1: To remove the small details, dilate the image then erode
Step 2: Compute the dissimilarity between the obtained and the original one
Step 3: To remove noise first dilate and then the mask of difference is eroded
Step 4: A Boolean mask is created for artifacts location; and
Step 5: Replace the mask covered pixels by original image's pixel

Figure 4: Dull Razor Algorithm

CONS

With modern techniques the primary disadvantage is that it's a very technical domain understood by very few people and thus it is difficult to execute on. Importantly, every modern ML model shipped by a major company in the space (i.e., Google, FB, OpenAI) relies on transfer learning at its core.

TRANSFER LEARNING (MOST COMMON)

Transfer learning in a modern context requires a very large, general dataset. It's extremely important that you not build your base model from domain-specific data. If you build your base model on domain-specific data (god forbid data in your training or test sets) then your entire experiment is invalid. Even if you don't present labels to the base model, by seeing the "correct" base data you've given it information that it shouldn't have access to.

Transfer learning is the reuse of a pre-trained model on a new problem. It's currently very popular in deep learning because it can train deep neural networks with comparatively little data.

In transfer learning, the knowledge of an already trained machine learning model is applied to a different but related problem.

With transfer learning, we basically try to exploit what has been learned in one task to improve generalization in another. We transfer the weights that a network has learned at "task A" to a new "task B."

The general idea is to use the knowledge a model has learned from a task with a lot of available labelled training data in a new task that doesn't have much data. Instead of starting the learning process from scratch, we start with patterns learned from solving a related task.

USING A PRE-TRAINED MODEL

The second approach is to use an already pre-trained model. There are a lot of these models out there, so make sure to do a little research. How many layers to reuse and how many to retrain depends on the problem.

FEATURE EXTRACTION

Another approach is to use deep learning to discover the best representation of your problem, which means finding the most important features. This approach is also known as representation learning, and can often result in a much better performance than can be obtained with hand-designed representation.

PROS

Transfer learning has several benefits, but the main advantages are saving training time, better performance of neural networks (in most cases), and not needing a lot of data. Additionally, training time is reduced because it can sometimes take days or even weeks to train a deep neural network from scratch on a complex task.

- Hyper-parameters for Image Augmentation and CNNs (common)
    - CNN to Binary Output

ADAM

PROS

- Easy to implement.
- Quite computationally efficient.
- Requires little memory space.
- Good for non-stationary objectives.
- Works well on problems with noisy or sparse gradients.
- Works well with large data sets and large parameters.

CONS

There are few disadvantages as the Adam optimizer tends to converge faster, but other algorithms like the Stochastic gradient descent focus on the datapoints and generalize in a better manner. Thus, the performance depends on the type of data being provided and the speed/generalization trade-off.

PROPERTIES OF ADAM

Here I list some of the properties of Adam, for proof that these are true refer to the paper:

1. Actual step size taken by the Adam in each iteration is approximately bounded the step size hyper-parameter. This property adds intuitive understanding to previous unintuitive learning rate hyper-parameter.
2. Step size of Adam update rule is invariant to the magnitude of the gradient, which helps a lot when going through areas with tiny gradients (such as saddle points or ravines). In these areas SGD struggles to quickly navigate through them.
3. Adam was designed to combine the advantages of Adagrad, which works well with sparse gradients, and RMSprop, which works well in on-line settings. Having both of these enables us to use Adam for broader range of tasks. Adam can also be looked at as the combination of RMSprop and SGD with momentum.

PROBLEMS WITH ADAM

When Adam was first introduced, people got very excited about its power. Paper contained some very optimistic charts, showing huge performance gains in terms of speed of training

RMSPROP

The RMSProp algorithm full form is called **Root Mean Square Prop**, which is an adaptive learning rate optimization algorithm proposed by Geoff Hinton.

RMSProp has several advantages; for one, it is a very robust optimizer which has pseudo curvature information. Additionally, it can deal with stochastic objectives very nicely, making it applicable to mini batch **learning**. Works with gnumpy. (float or array_like) Step rate of the optimizer.

$$m_t = \beta_1 m_{t-1} + (1-\beta_1) g_t$$
$$v_t = \beta_2 v_{t-1} + (1-\beta_2) g_t^2$$

Moving averages of gradient and squared gradient.

Figure 5: RMSProp Formulae

$$E[m_t] = E[g_t]$$
$$E[v_t] = E[g_t^2]$$

Figure 6: RMSProp Formulae

DCNN

A **deep convolutional neural network** (**DCNN**) consists of many neural network layers. Two different types of layers, convolutional and pooling, are typically alternated. The depth of each filter increases from left to right in the network. The last stage is typically made of one or more fully connected layers:

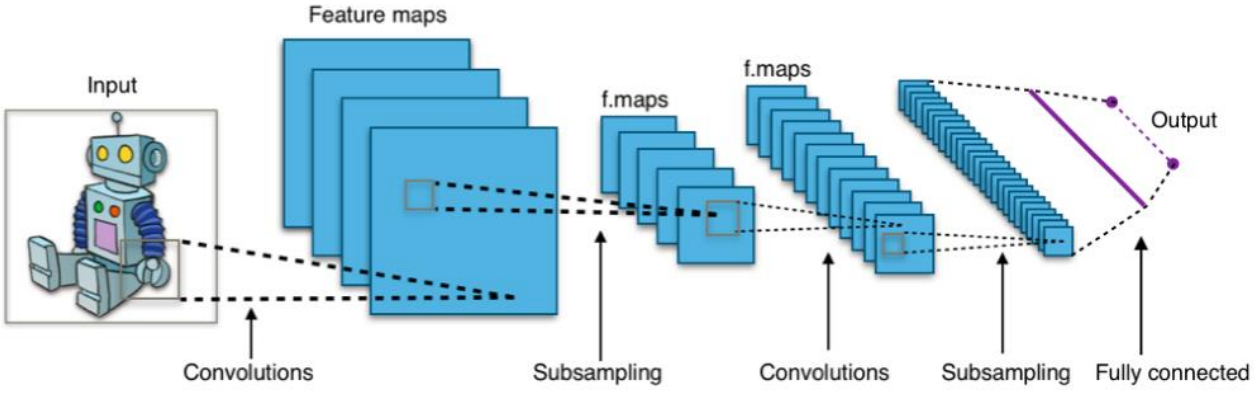

Figure 7: DCNN Architecture

There are three key intuitions beyond ConvNets:

- Local receptive fields
- Shared weights
- Pooling

PROS

CNNs is **weight sharing**. Let's take an example to explain this. Say you have a one layered CNN with 10 filters of size 5x5. Now you can simply calculate parameters of such a CNN, it would be 5*5*10 weights and 10 biases i.e., **5* 5*10 + 10 = 260 parameters**. Now let's take a simple one layered NN with 250 neurons, here the number of weight parameters depending on the size of images is '250 x K' where size of the image is P X M and K = (P *M). Additionally, you need 'M' biases. For the MNIST data as input to such a NN we will have (**250*784+1 = 19601) parameters.** Clearly, CNN is more efficient in terms of memory and complexity.

In terms of performance, CNNs outperform NNs on conventional image recognition tasks and many other tasks. Look at the Inception model, Resnet50 and many others for instance.

For a completely new task / problem CNNs are very good **feature extractors.** This means that you can extract useful attributes from an already trained CNN with its trained weights by feeding your data on each level and tune the CNN a bit for the specific task.

CONS

- CNN do not encode the position and orientation of object
- Lack of ability to be spatially invariant to the input data
- ANN, SVM, Naïve-Bayes Algorithm

ONE-HOT ENCODING

PROS

One-hot encoding ensures that machine learning does not assume that higher numbers are more important. For example, the value '8' is bigger than the value '1', but that does not make '8' more important than '1'.

CONS

The disadvantage is that for high cardinality, the feature space can really blow up quickly and you start fighting with the curse of dimensionality.

NOISE REMOVAL

Noise removal algorithm is the process of removing or reducing the noise from the image. The noise removal algorithms reduce or remove the visibility of noise by smoothing the entire image leaving areas near contrast boundaries.
One of the most popular methods is wiener filter. In this work four types of noise (Gaussian noise, Salt & Pepper noise, Speckle noise and Poisson noise) is used and image de-noising performed for different noise by Mean filter, Median filter and Wiener filter. Further results have been compared for all noises.

The noise degrades performance of image processing algorithms in brain imaging. Image denoising methods are important image processing algorithms which are used to reduce the noise.

The noise degrades performance of image processing algorithms in brain imaging. Image denoising methods are important image processing algorithms which are used to reduce the noise:

- Segmentation

- Resize, Feature Extraction, Classification (Common)

ON THE BASIS OF FEATURE SELECTION TECHNIQUES

Some of the feature selection methods can be grouped as:

FEATURE SELECTION METHODS:

I will share 3 Feature selection techniques that are easy to use and also gives good results.

1. Univariate Selection
2. Feature Importance
3. Correlation Matrix with Heatmap

1. UNIVARIATE SELECTION

Statistical tests can be used to select those features that have the strongest relationship with the output variable. The scikit-learn library provides the SelectKBest class that can be used with a suite of different statistical tests to select a specific number of features.

2. FEATURE IMPORTANCE

You can get the feature importance of each feature of your dataset by using the feature importance property of the model.

Feature importance gives you a score for each feature of your data, the higher the score more important or relevant is the feature towards your output variable.

Feature importance is an inbuilt class that comes with Tree Based Classifiers, we will be using Extra Tree Classifier for extracting the top 10 features for the dataset.

3. CORRELATION MATRIX WITH HEATMAP

Correlation states how the features are related to each other or the target variable.

Correlation can be positive (increase in one value of feature increases the value of the target variable) or negative (increase in one value of feature decreases the value of the target variable)

Heatmap makes it easy to identify which features are most related to the target variable, we will plot heatmap of correlated features using the seaborn library.

There are two main types of feature selection techniques: supervised and unsupervised, and supervised methods may be divided into wrapper, filter and intrinsic:

- Filter-based feature selection methods use statistical measures to score the correlation or dependence between input variables that can be filtered to choose the most relevant features.
- Statistical measures for feature selection must be carefully chosen based on the data type of the input variable and the output or response variable

WAYS OF FEATURE SELECTION

Select a subset of input features from the dataset, then:

- **Unsupervised**: Do not use the target variable (e.g., remove redundant variables).
  - Correlation
- **Supervised**: Use the target variable (e.g., remove irrelevant variables).
  - **Wrapper**: Search for well-performing subsets of features.
    - RFE
  - **Filter**: Select subsets of features based on their relationship with the target.
    - Statistical Methods
    - Feature Importance Methods
  - **Intrinsic**: Algorithms that perform automatic feature selection during training.
    - Decision Trees

TYPES OF FEATURE SELECTION

There are three **types of feature selection**: Wrapper methods (forward, backward, and stepwise **selection**), Filter methods (ANOVA, Pearson correlation, variance thresholding), and Embedded methods (Lasso, Ridge, Decision Tree).

PEARSON CORRELATION

PROS

Pearson's Correlation Coefficient helps you find out the relationship between two quantities. It gives you the measure of the strength of association between two variables. The value of **Pearson's Correlation Coefficient** can be between -1 to +1. 1 means that they are highly **correlated** and 0 means no **correlation**.

CONS

The **disadvantages** of the **Pearson** r **correlation** method are:

- It assumes that there is always o linear relationship between the variables which might not be the case at all times
- It can be easily misinterpreted as a high degree of **correlation** from large values of the **correlation coefficient**.

DECISION TREE

Decision Tree is a very popular machine learning algorithm. Decision Tree solves the problem of machine learning by transforming the data into a tree representation. Each internal node of the tree representation denotes an attribute and each leaf node denotes a class label.

A decision tree algorithm can be used to solve both regression and classification problems.

PROS

- Compared to other algorithms decision trees requires less effort for data preparation during pre-processing.
- A decision tree does not require normalization of data.

- A decision tree does not require scaling of data as well.
- Missing values in the data also do NOT affect the process of building a decision tree to any considerable extent.
- A Decision tree model is very intuitive and easy to explain to technical teams as well as stakeholders.

CONS

- A small change in the data can cause a large change in the structure of the decision tree causing instability.
- For a Decision tree sometimes, calculation can go far more complex compared to other algorithms.
- Decision tree often involves higher time to train the model.
- Decision tree training is relatively expensive as the complexity and time has taken are more.
- The Decision Tree algorithm is inadequate for applying regression and predicting continuous values.

RIDGE REGRESSION

Regularized methods such as Ridge Regression can be used to select only relevant features in the training dataset. The process of transforming a dataset in order to select only relevant features necessary for training is called dimensionality reduction.

PROS

We can use a regularized model to reduce the dimensionality of the training dataset. Dimensionality reduction is important because of three main reasons:

- **Prevents Overfitting**: A high-dimensional dataset having too many features can sometimes lead to overfitting (model captures both real and random effects).
- **Simplicity**: An over-complex model having too many features can be hard to interpret especially when features are correlated with each other.
- **Computational Efficiency**: A model trained on a lower dimensional dataset is computationally efficient (execution of algorithm requires less computational time).

CONS

- Regularization leads to dimensionality reduction, which means the machine learning model is built using a lower dimensional dataset. This generally leads to a high **bias error**.
- If regularization is performed before training the model, a perfect balance between **bias-variance trade-off** must be used.

LASSO

PROS

- As any regularization method, it can avoid overfitting. It can be applied even when number of features is larger than amount of data.
- It can do feature selection.
- It is fast in terms of inference and fitting.

CONS

- The model selected by lasso is not stable. For example, on different bootstrapped data, the feature selected can be very different.
- The model selection result is not intuitive to interpret: for example, why lasso select a feature?
- When there are highly correlated features, lasso may randomly select one of them of part of them. The result depends on the implementation. To improve, people introduced elastic net.
- Based on my experience, its prediction performance is usually worse than ridge regression in terms of MSE.

CLASSIFICATION

- CNN, Pooling Layer (common)
  - Max pooling (most common)
  - Sum pooling
  - Average pooling
- Autoencoders
  - Stacked Deep Autoencoders
- No. of Hidden layers in Dense and Sparse network
- Multilayer Perceptron
- Algorithms
  - SGNN
  - Genetic
  - Skin Lesion Segmentation'

DIFFERENCE BETWEEN SOFTMAX FUNCTION AND SIGMOID FUNCTION

SOFTMAX CLASSIFIER

SoftMax extends this idea into a multi-class world. That is, SoftMax assigns decimal probabilities to each class in a multi-class problem. Those decimal probabilities must add up to 1.0. This additional constraint helps training converge more quickly than it otherwise would.

Consider the following variants of SoftMax:

- **Full SoftMax** is the SoftMax we've been discussing; that is, SoftMax calculates a probability for every possible class.
- **Candidate sampling** means that SoftMax calculates a probability for all the positive labels but only for a random sample of negative labels. For example, if we are interested in determining whether an input image is a beagle or a bloodhound, we don't have to provide probabilities for every non-doggy example.

Full SoftMax is fairly cheap when the number of classes is small but becomes prohibitively expensive when the number of classes climbs. Candidate sampling can improve efficiency in problems having a large number of classes.

PROPERTIES OF SOFTMAX FUNCTION

Below are the few properties of SoftMax function.

- The calculated probabilities will be in the range of 0 to 1.
- The sum of all the probabilities is equals to 1.

SOFTMAX FUNCTION USAGE

- Used in multiple classification logistic regression model.
- In building neural networks SoftMax functions used in different layer level.

SIGMOID FUNCTION

PROPERTIES OF SIGMOID FUNCTION

- The sigmoid function returns a real-valued output.
- The first derivative of the sigmoid function will be non-negative or non-positive.
    - **Non-Negative:** If a number is greater than or equal to zero.
    - **Non-Positive:** If a number is less than or equal to Zero.

SIGMOID FUNCTION USAGE

- The Sigmoid function used for **binary classification** in logistic regression model.
- While creating artificial neuron's sigmoid function used as the **activation function**.
- In statistics, the **sigmoid function graphs** are common as a cumulative distribution function

SOFTMAX FUNCTION VS SIGMOID FUNCTION

While learning the logistic regression concepts, the primary confusion will be on the functions used for calculating the probabilities. As the calculated probabilities are used to predict the target class in logistic regression model. The two principal functions we frequently hear are SoftMax and Sigmoid function.

Even though both the functions are same at the **functional level.** (Helping to predict the target class) many noticeable mathematical differences are playing the vital role in using the functions in deep learning and other fields of areas.

SoftMax function calculates the probabilities distribution of the event over 'n' different events. In general way of saying, this function will calculate the probabilities of each target class over all possible target classes. Later the calculated probabilities will be helpful for determining the target class for the given inputs.

The main advantage of using SoftMax is the output probabilities range. The range will **0 to 1**, and the sum of all the probabilities will be **equal to one**. If the SoftMax function used for multi-classification model it returns the probabilities of each class and the target class will have the high probability.

BINARY CLASSIFIER

It is one of the most frequently used problems in machine learning. In simplest form the user tries to classify a unit into 1 of the 2 possible categories. For example, take the attributes of the fruits like color, peel texture, shape etc. A linear classifier that the perceptron is classified as is a classification algorithm, which depends on a linear predictor function to make the predictions and predictions are based on the union that includes weights and feature vector.

DENSE LAYER

Neural network dense layers (or fully connected layers) are the foundation of nearly all neural networks.

THE PROBLEM WITH THE PERCEPTRON

Neural networks come in many different variations these days, from convolutional and recurrent, to homogenous and heterogeneous, to linear and branching.

But the original neural networks were a single neuron: the perceptron. Perceptron's showed some promise, but came up short when attempting to handle some of the simplest logical operations. Unfortunately, perceptron didn't have enough complexity to approximate many of the functions that neural networks can approximate today.

THE SOLUTION WAS TO ADD MORE NEURONS

WHAT IS MULTILAYER PERCEPTRON?

The perceptron is very useful for classifying data sets that are linearly separable. They encounter serious limitations with data sets that do not conform to this pattern as discovered with the XOR problem. The XOR problem shows that for any classification of four points that there exists a set that are not linearly separable.

The algorithm for the MLP is as follows:

1. Just as with the perceptron, the inputs are pushed forward through the MLP by taking the dot product of the input with the weights that exist between the input layer and the hidden layer (WH). This dot product yields a value at the hidden layer. We do not push this value forward as we would with a perceptron though.
2. MLPs utilize activation functions at each of their calculated layers. There are many activation functions to discuss: rectified linear units (ReLU), sigmoid function, tanh. Push the calculated output at the current layer through any of these activation functions.
3. Once the calculated output at the hidden layer has been pushed through the activation function, push it to the next layer in the MLP by taking the dot product with the corresponding weights.
4. Repeat steps two and three until the output layer is reached.
5. At the output layer, the calculations will either be used for a backpropagation algorithm that corresponds to the activation function that was selected for the MLP (in the case of training) or a decision will be made based on the output (in the case of testing).

GLCM

The **GLCM functions** characterize the texture of an image by calculating how often pairs of pixels with specific values and in a specified spatial relationship occur in an image, creating a **GLCM**, and then extracting statistical measures from this matrix. GLCM set of features are based on second order statistics... they can be used to reflect, the overall average for degree of correlation between pairs of pixels in different aspects (in terms of homogeneity, uniformity...etc.). One of the main factors affects that affects the discrimination capabilities of GLCM is the separation distance between pixels... When you take the distance 1 it leads to reflect the degree of correlation between adjacent pixels (i.e., short range neighborhood connectivity). While, increasing the distance value leads to reflect the degree of correlation between distant pixels

ON THE BASIS OF MODEL TRAINING METHOD

Some of the training methods can be grouped as:

- MVSM
- Transfer Learning: (used as a group of 2-3 with CNN)
    - **Inception (V1, V3)**

- ResNet
- ResNet50
- MobileNet

- Deep Pipeline

MCNN

Traditional approaches typically involve extracting discriminative features from the original time series using dynamic time warping (DTW) or shape transformation, based on which an off-the-shelf classifier can be applied. These methods are ad-hoc and separate the feature extraction part with the classification part, which limits their accuracy performance. Plus, most existing methods fail to take into account the fact that time series often have features at different time scales. To address these problems, we propose a novel end-to-end neural network model, Multi-Scale Convolutional Neural Networks (MCNN), which incorporates feature extraction and classification in a single framework. Leveraging a novel multi-branch layer and learnable convolutional layers, MCNN automatically extracts features at different scales and frequencies, leading to superior feature representation. MCNN is also computationally efficient, as it naturally leverages GPU computing. MCNN advances the state-of-the-art by achieving superior accuracy performance than other leading methods.

CNN (MOST COMMON)

PROS

In terms of architecture, the key building block of CNN is the convolutional layer. According to a MathWorks post, a CNN convolves learned features with input data, and uses 2D convolutional layers, making this architecture well suited to processing 2D data, such as images. Since CNNs eliminate the need for manual feature extraction, one doesn't need to select features required to classify the images. How CNN work is by extracting features directly from images and the key features are not pretrained; they are learned while the network trains on a collection of images, the post notes. It is the automated feature extraction that makes CNNs highly suited for and accurate for computer vision tasks such as object/image classification.

CONS

Convolutional neural networks like any neural network model are computationally expensive. But that is more of a drawback than a weakness. This can be overcome with better computing hardware such as GPUs and Neuromorphic chips.

GOOGLENET/INCEPTION

While VGG achieves a phenomenal accuracy on ImageNet dataset, its deployment on even the most modest sized GPUs is a problem because of huge computational requirements, both in terms of memory and time. It becomes inefficient due to large width of convolutional layers.

For instance, a convolutional layer with 3X3 kernel size which takes 512 channels as input and outputs 512 channels, the order of calculations is 9X512X512.

In a convolutional operation at one location, every output channel (512 in the example above), is connected to every input channel, and so we call it a dense connection architecture. The GoogLeNet builds on the idea that most of the activations in a deep network are either unnecessary (value of zero) or redundant because of correlations between them. Therefore, the most efficient architecture of a deep network will have a sparse connection between the

activations, which implies that all 512 output channels will not have a connection with all the 512 input channels. There are techniques to prune out such connections which would result in a sparse weight/connection. But kernels for sparse matrix multiplication are not optimized in BLAS or CuBlas (CUDA for GPU) packages which render them to be even slower than their dense counterparts.

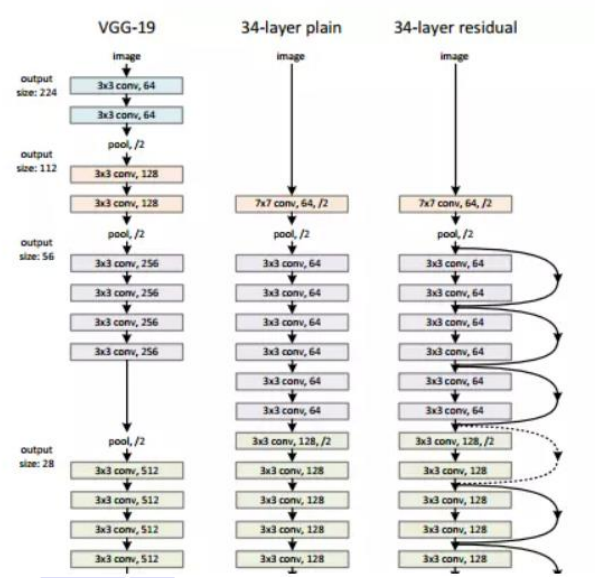

Figure 8: GoogLeNet/Inception Architecture

ALEXNET

This architecture was one of the first deep networks to push ImageNet Classification accuracy by a significant stride in comparison to traditional methodologies. It is composed of 5 convolutional layers followed by 3 fully connected layers.

AlexNet, proposed by Alex Krizhevsky, **uses ReLU (Rectified Linear Unit) for the non-linear part, instead of a Tanh or Sigmoid function which was the earlier standard for traditional neural networks**. ReLU is given by

$$f(x) = \max(0, x)$$

PROS

The advantage of the ReLU over sigmoid is that it trains much faster than the latter because the derivative of sigmoid becomes very small in the saturating region and therefore the updates to the weights almost vanish. This is called **vanishing gradient problem**.

In the network, ReLU layer is put after each and every convolutional and fully-connected layers (FC).

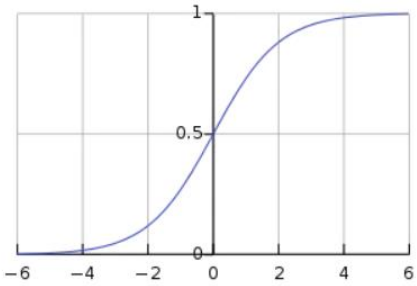

Figure 9: ReLU Function Graph

Another problem that this architecture solved was reducing the **over-fitting** by using a Dropout layer after every FC layer. Dropout layer has a probability, **(p)**, associated with it and is applied at every neuron of the response map separately. It randomly switches off the activation with the probability **p**

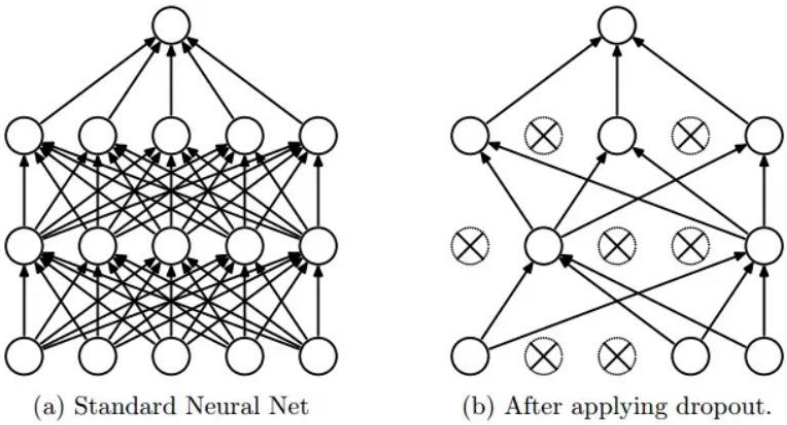

Figure 10: ReLU Function Net

WHY DOES DROPOUT WORK?

The idea behind the dropout is similar to the model ensembles. Due to the dropout layer, different sets of neurons which are switched off, represent a different architecture and all these different architectures are trained in parallel with weight given to each subset and the summation of weights being one. For n neurons attached to Dropout, the number of subset architectures formed is 2^n. So, it amounts to prediction being averaged over these ensembles of models. This provides a structured model regularization which helps in avoiding the over-fitting. Another view of Dropout being helpful is that since neurons are randomly chosen, they tend to avoid developing co-adaptations among themselves thereby enabling them to develop meaningful features, independent of others.

VGG16

This architecture is from VGG group, Oxford. It makes the improvement over AlexNet by replacing large kernel-sized filters (11 and 5 in the first and second convolutional layer, respectively) with multiple 3X3 kernel-sized filters one after another. With a given receptive field (the effective area size of input image on which output depends),

multiple stacked smaller size kernel is better than the one with a larger size kernel because multiple non-linear layers increase the depth of the network which enables it to learn more complex features, and that too at a lower cost.

For example, three 3X3 filters on top of each other with stride 1 ha a receptive size of 7, but the number of parameters involved is 3*(9C^2) in comparison to 49C^2 parameters of kernels with a size of 7. Here, it is assumed that the number of input and output channel of layers is C. Also, 3X3 kernels help in retaining finer level properties of the image. The network architecture is given in the table.

You can see that in VGG-D, there are blocks with same filter size applied multiple times to extract more complex and representative features. This concept of blocks/modules became a common theme in the networks after VGG.

The VGG convolutional layers are followed by 3 fully connected layers. The width of the network starts at a small value of 64 and increases by a factor of 2 after every sub-sampling/pooling layer. It achieves the top-5 accuracy of 92.3 % on ImageNet.

XCEPTION

Xception is a deep convolutional neural network architecture that involves Depth wise Separable Convolutions.

WHAT IS AN XCEPTION NETWORK?

The data first goes through the entry flow, then through the middle flow which is repeated eight times, and finally through the exit flow. Note that all Convolution and Separable Convolution layers are followed by batch normalization

HOW DOES XCEPTION WORK?

Xception is an efficient architecture that relies on two main points:

- Depth wise Separable Convolution
- Shortcuts between Convolution blocks as in ResNet

IMPLEMENTATION OF THE XCEPTION

Xception offers an architecture that is made of Depth wise Separable Convolution blocks + MaxPooling, all linked with shortcuts as in ResNet implementations.

The specificity of Xception is that the Depth wise Convolution is not followed by a Pointwise Convolution.

THE LIMITS OF CONVOLUTIONS:

First of all, Convolution is a really expensive operation. To overcome the cost of such operations, depth wise separable convolutions have been introduced. They are themselves divided into 2 main steps:

- Depth wise Convolution
- Pointwise Convolution

SGD

PROS

- Memory requirement is less compared to the GD algorithm as derivative is computed taking only 1 point at once.

CONS

- The time required to complete 1 epoch is large compared to the GD algorithm.
- Takes a long time to converge.
- May stuck at local minima.

ON THE BASIS OF MODEL EVALUATION METHOD

Some of the evaluation methods can be grouped as:

- Most Used:
  - Precision & Recall
  - F-score
  - Accuracy (Most Common)
  - Random Forest, XGBoost, SVM
  - Specificity & Sensitivity
  - Confusion Matrix
  - ABCDE Criteria
- Rare:
  - Jaccard similarity coefficient (JSC)
  - geometric mean (G-mean)
  - Matthew's correlation coefficient (MCC)
  - Cohen's kappa score (CKS)
  - AUROC
  - precision-recall curve (PR-AUC)
  - evaluation time

FUTURE SCOPE

- Implementation of various other algorithms and using several optimization techniques. Also, more data will be collected in order to recognize the features more accurately.
- Major attention will be given to increase the accuracy such that our proposed system can be used to detect a large number of chronic and critical diseases.
- When these enhancements are done, the system can be integrated with an android application to make it more convenient and easily portable. This will allow people from all strata to use it effectively even if they do not have a personal computer.

CONCLUSIONS

- A "health discernment system" has been proposed for medical image classification that will work in real-life scenarios.
- The proposed method is based on ***Convolutional Neural Network*** architecture.
- Different sub-models pertaining to the two diseases (skin cancer: Melanoma, Benign) have been designed using convolutional neural network (CNN) and they have all been tested separately.
- For pre-Processing method, we have come to transfer learning as the best algorithm as it works best with most neural networks, doesn't require a lot of data and consumes very less time as compared to others

- For Feature selection method, we have arrived at correlation matrix with heatmap as the heatmap makes it easy to identify which features are most related to the target variable and it can be trained under supervised and unsupervised methods with three subclasses under each method.
- For model selection or the activation function we think the SoftMax function is the best option as it used in multiple classification logistic regression model and in building neural networks SoftMax functions used in different layer level. SoftMax assigns decimal probabilities to each class in a multi-class problem. Those decimal probabilities must add up to 1.0. This additional constraint helps training converge more quickly than it otherwise would.
- For model training the best method out of all the listed one's is the CNN algorithm as a CNN convolves learned features with input data, and uses 2D convolutional layers, making this architecture well suited to processing 2D data, such as images and also CNNs highly suited for and accurate for computer vision tasks such as object/image classification.

REFERENCES

CONCEPTS

- ResearchGate.com
- ScienceDirect.com
- GeeksforGeeks.com
- TutorialsPoint.com
- cs.stanford.edu
- Springer.com
- towardsdatascience.com
- academictorrents.com
- kaagle.com

RESEARCH

- https://www.researchgate.net/publication/334751850_Skin_Cancer_Detection_Using_Convolutional_Neural_Network
- Peer-review under responsibility of the scientific committee of the 16th International Learning & Technology Conference 2019. 10.1016/j.procs.2019.12.090
- https://thesai.org/Publications/IJACSA
- https://www.thelancet.com/journals/ebiom/article/PIIS2352-3964(19)30294-4/fulltext
- https://www.sciencedirect.com/science/article/pii/S2215016120300832?via%3Dihub
- https://www.sciencedirect.com/science/article/pii/S1532046418301618?via%3Dihub
- https://www.ejcancer.com/article/S0959-8049(19)30381-8/fulltext
- https://www.researchgate.net/publication/335321267_Bio-Inspired_DeepCNN_Pipeline_for_Skin_Cancer_Early_Diagnosis
- https://www.researchgate.net/publication/347927500_Automated_Multiclass_Classification_of_Skin_Lesions_through_Deep_Convolutional_Neural_Network_with_Dermoscopic_Images
- https://www.researchgate.net/publication/339804392_Deep_Learning_Solutions_for_Skin_Cancer_Detection_and_Diagnosis?enrichId=rgreq-cc1ddabe639ba17298c595c27b48021c-XXX&enrichSource=Y292ZXJQYWdlOzMzOTgwNDM5MjtBUzo4ODM0MjE3NjUxNzMyNTFAMTU4NzYzNTU3MDg4Nw%3D%3D&el=1_x_3&_esc=publicationCoverPdf
- https://ieeexplore.ieee.org/document/9062473
- https://www.sciencedirect.com/science/article/pii/S0933365719301460
- https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3352407

- https://www.researchgate.net/publication/335337495_A_CNN_toolbox_for_skin_cancer_classification?enrichId=rgreq-91f014447a4c29cec1b8c22e6f8930fb-XXX&enrichSource=Y292ZXJQYWdlOzMzNTMzNzQ5NTtBUzo4MDExNDUyMDMyMjQ1NzdAMTU2ODAxOTMwOTE5Nw%3D%3D&el=1_x_3&_esc=publicationCoverPdf
- https://ieeexplore.ieee.org/document/8720210
- https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6231861/
- https://iopscience.iop.org/article/10.1088/1757-899X/982/1/012005
- https://www.researchgate.net/publication/327539277_Skin_Cancer_Classification_using_Convolutional_Neural_Networks_Systematic_Review_Preprint
- https://www.sciencedirect.com/science/article/pii/S2352914819302047
- https://www.researchgate.net/publication/334123580_Melanoma_Skin_Cancer_Detection_using_Image_Processing_and_Machine_Learning
- https://www.nature.com/articles/nature21056
- https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=9113301
- https://pubmed.ncbi.nlm.nih.gov/28117445/
- https://pubmed.ncbi.nlm.nih.gov/30333097/
- https://ieeexplore.ieee.org/document/9198489