Introduction to Artificial Intelligence
2020/21
Coursework

Classifying Skin Lesion Data using Convolutional Neural Networks and Random Forests

Tommaso Capecchi
Vittoria Castelnuovo

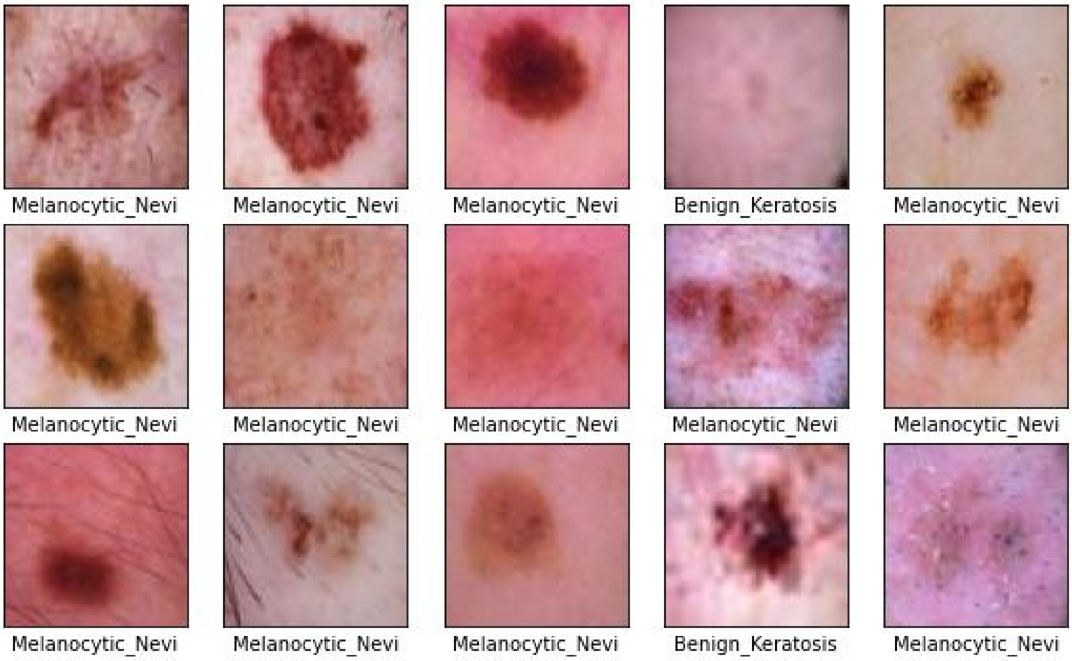


Table of Contents:

Introduction:

Artificial intelligence techniques have been adopted in the medical field as a way to take out human error, accelerate manual processes, and as a way to predict possible diagnoses. In this project we aim to create a classifier system which discriminates between various images of skin lesions, abrasion, and cancers. The end goal is to create a model which is able to differentiate and identify between different type of malformations.

Before technology was adopted into the medical realm, these types of skin abrasions were previously only detected by industry professionals with surplus of experience, as well as medical training. Histologic exams are the most common way for medical professionals to classify these skin abrasions. This process involves a biopsy on a sample of the patient’s malformed skin, which is then analyzed in detail (Esteva et al., 2017). For a convincing diagnosis to be made, a dermatologist must take into consideration various factors, each of which are specific to the patient. These factors may include, but are not limited to, family history of skin cancer, possible underlying health conditions, and exposure to ultraviolet (UV) light. Moreover, a dermatologist must be knowledgeable about many different skin conditions, in order to recognize them. For instance, the irregularity of a skin lesion may be a weighted indicator of a possible malformation of cells.

Due to the case-by-case nature of these diagnoses, the detection and treatment of these skin conditions may be delayed, thus impacting the recovery of the patient, and in some cases aggravating the progression of their condition. The early detection and classification of these skin lesions is extremely important, as the onset of these dermatologic abnormalities can progress quite rapidly and lead, in some cases, to fatality.

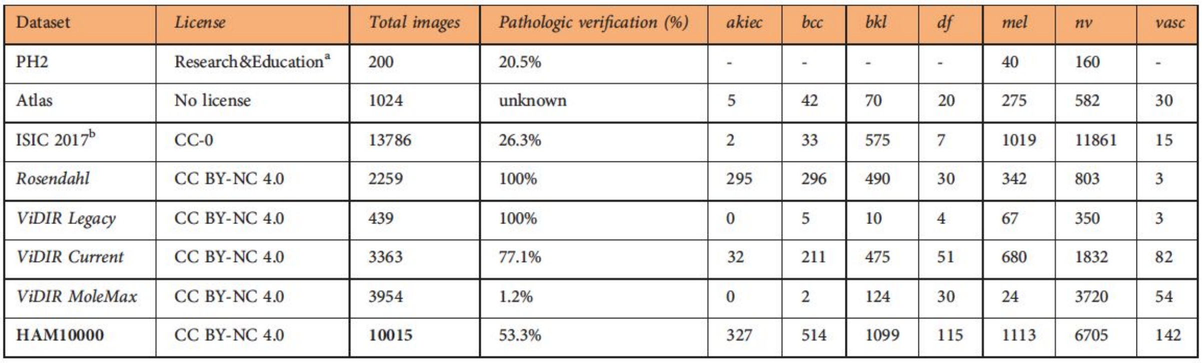
This project aims to aid this endeavor by creating two systems, namely a Convolutional Neural Network (CNN) and a Random Forest (RF) which analyze medical data and correctly classify each type of skin lesion. The broader aim of this project is to shorten the time of diagnosis, so that skin condition treatments are given more promptly to patients, as there is a higher chance of recovery if these malformations are detected in their initial stages.

We hypothesize that the CNN will detect and classify the data more accurately, as these networks have garnered a state-of-the-art level of performance on image recognition and classification tasks. Alternatively, we hypothesize that the RF will perform well overall, although there may be some discrepancies on generalization, as the dataset is biased towards one class.

Dataset Analysis:

The dataset selected for this project is the Skin Cancer: HAM10000 (“Humans Against Machines with 10000 training images”), which can be found on Harvard’s dataverse website (dataverse.harvard.com) (Harvard Dataverse, 2018). The dataset contains more than 10000 600x450 pigmented images of varying types of skin lesions, including: Bowen's disease, basal cell carcinoma, benign keratosis-like lesions, dermatofibroma, melanoma, melanocytic nevi, and vascular lesions. A Comma Separated Values (CSV) file containing the images’ metadata is located alongside this dataset.

This dataset was chosen for a variety of reasons. Firstly, when creating machine learning applications, it is important to choose data which reflects the real world. In order to implement a solution to a real-life problem, the data used to train the algorithm must fit the scenarios in which the problem itself can be encountered. The HAM10000 dataset reflects this requirement as the images are of high quality and have been adapted with several optimization techniques (reshaped, color and brightness correction, centered, etc…) to produce data that is well formed for the scope of the classification (Harvard Dataverse, 2018).



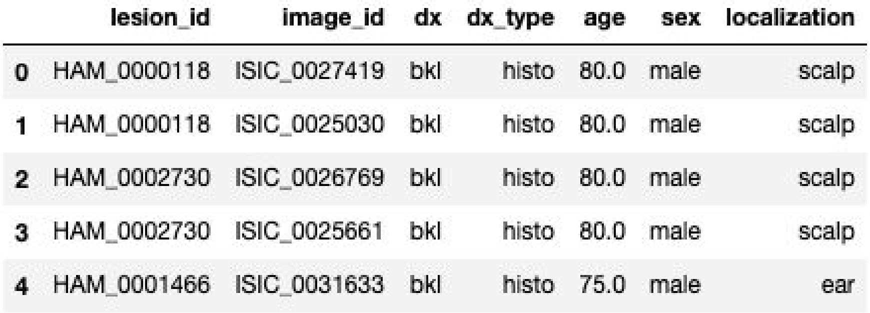
Dataset	*License*	*Total images*	*Pathologic verification (%)*	*akiec*	*bcc*	*bkl*	*df*	*mel*	*nv*	*vasc*
PH2	Research&Education[a]	200	20.5%	-	-	-	-	40	160	-
Atlas	No license	1024	unknown	5	42	70	20	275	582	30
ISIC 2017[b]	CC-0	13786	26.3%	2	33	575	7	1019	11861	15
Rosendahl	CC BY-NC 4.0	2259	100%	295	296	490	30	342	803	3
ViDIR Legacy	CC BY-NC 4.0	439	100%	0	5	10	4	67	350	3
ViDIR Current	CC BY-NC 4.0	3363	77.1%	32	211	475	51	680	1832	82
ViDIR MoleMax	CC BY-NC 4.0	3954	1.2%	0	2	124	30	24	3720	54
HAM10000	CC BY-NC 4.0	**10015**	53.3%	327	514	1099	115	1113	6705	142

As the image above details, the HAM10000 dataset appears to be the most complete medical database for skin cancer classification as the data is abundant and verified via a number of pathological examinations.

Additionally, having images and as well as metadata is a great benefit as it facilitates data manipulation. Different modalities of data promote creative ways of shaping, manipulating and evaluating it. Moreover, the two models chosen in this project evaluate the dataset in different ways: the neural network convolves each image attempting to detect patterns and discriminate features for each class. On the other hand, the random forest uses the metadata as a decision boundary for its splitting. These two models both have their unique advantages and portray different benefits to the same classification task. For instance, in particular cases where only metadata is available, the random forest model may be preferred; whereas, when dermatoscopic images are available, CNNs may fit the specific needs of the task at hand.

Data processing & Augmentation:

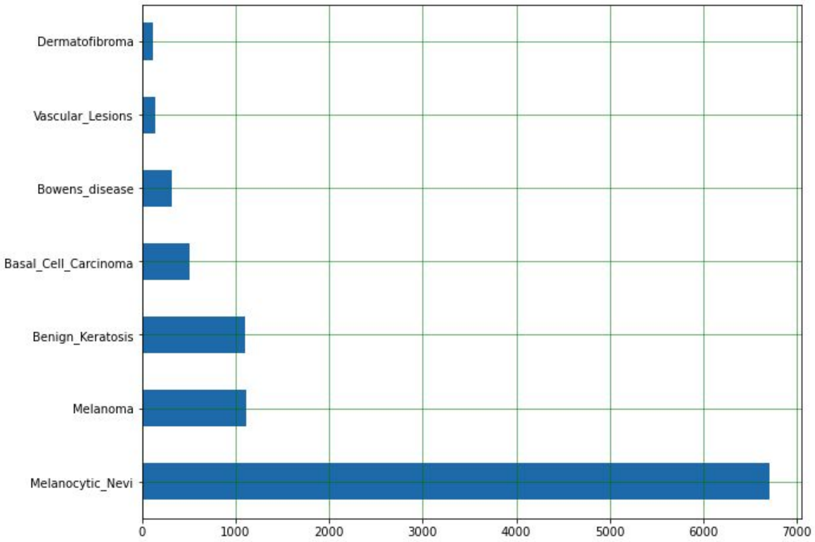
Initially, the Pandas framework was employed as the main framework to analyse and process the data. The most relevant features are 'dx', 'dx_type' and 'localization', which indicate the type of skin cancer, the medical exam performed for diagnosis, and the surface where the lesion is located on the body, respectively.



	lesion_id	image_id	dx	dx_type	age	sex	localization
0	HAM_0000118	ISIC_0027419	bkl	histo	80.0	male	scalp
1	HAM_0000118	ISIC_0025030	bkl	histo	80.0	male	scalp
2	HAM_0002730	ISIC_0026769	bkl	histo	80.0	male	scalp
3	HAM_0002730	ISIC_0025661	bkl	histo	80.0	male	scalp
4	HAM_0001466	ISIC_0031633	bkl	histo	75.0	male	ear

As some of the values in the 'age' column were missing, these were replaced with the median age, in order to avoid runtime errors. Moreover, the images were reshaped to a lower resolution, from 600x 450 pixels to 64x64, in order to accelerate the training phase without losing valuable information.

A setback of this dataset is its bias towards the melanocyte nevi class (as the graph below illustrates), which accounts for about half of the total images. In order for the classifiers to learn all classes uniformly, some artificial data which mimics the distribution each class must be created. By using Keras' ImageDataGenerator() class, images from the imbalanced classes were picked at random and reformatted, which produced artificial datapoints to converge with our pre-existing dataset. Finally, our dataset included 46890 images.



It is worth noting that the augmented data (approximately 37000 images) is used for the training phase exclusively, and the predictions will be made on the original (reshaped, but not augmented) images (10015 images).

Methodology:

Convolutional Neural Network:

The first machine learning model implemented for this project is a CNN. Introduced by Yann LeCun et al. in 1998, CNNs have proved to be extremely efficient supervised learning models for image classification and recognition, especially when employed for classification tasks. In fact, these types of networks work best when manipulating data with grid-like structures, such as images which can be structured as a “grid” of pixels.

CNNs are extremely efficient at extracting features because as images are parsed through filters, they transform into 3-dimensional volumes of the image itself with scanned by different kernels (Venkatesan and Li, 2017). This promotes parameter sharing within the network, as well as reduces the number of parameters to learn as the same filter is used for each pixel of the image, due to the network’s equivariant properties (Liew, Khalil-Hani and Bakhteri, 2016).

Building the CNN:

The structure of neural networks is inspired by the human brain where each neuron is connected to another through synapses which “fire” upon the detection of a specific stimuli (Merkulov, Yang and Zheng, 2018). The computational power of a neuron is quite limited, however, when many of these are combined together, they produce extremely powerful results.

CNNs are usually comprised of various layers: an input layer, an arbitrary number of hidden layers, and an output layer. The hidden layers of this network are the most essential component of this model, where learning happens. These hidden layers involve convolutional layers (which use kernels to “scan” through the image and detect features), pooling layers (which are used to capture the essence of the pixels in each segment of the image), activation layers (where an activation function is used to “fire” the perceptron depending on certain factors), and lastly a fully connected layer (which receives a flattened feature map as a one-dimensional vector and output a classification result).

The CNN model implemented in this project uses the Keras framework provided by Tensorflow, a module for the creation and manipulation of neural networks. This library is commonly used in the field of artificial intelligence as it is an opensource framework, which enables users to implement neural networks in a straightforward and high-level manner.

Training CNN:

Initially, each image was converted into a Numpy array of shape (64,64,3), a 64x64 pixel image with an RGB channel. Subsequently, each pixel was normalized to a value between 0 and 1, to ensure numerical stability while training, by divide the pixel value by 255. Using the Scikit-learn library, the images were partitioned into a training and testing set. The CNN model was developed with the following characteristics:

- First Convolutional layer of 16 filters, kernel size (3,3), strides=1
- Second Convolutional layer of 32 filters, kernel size (3,3), strides=1
- First Pooling layer, pool size of (2,2), strides=1
- Third Convolutional layer of 64 filters, kernel size (2,2), strides=1
- Fourth Convolutional layer of 64 filters, kernel size (2,2), strides=1
- Second Pooling layer, pool size (2,2), strides=1
- First hidden layer of fully connected NN of 128 neurons
- Second hidden layer of the NN of 64 neurons
- One Dropout layer with probability of 0.1

A softmax activation function for the output layer was deemed appropriate for this multiclassification task, while keeping the Rectified Linear Unit (ReLU) activation present in the convolutional layers. This activation function was chosen over the Sigmoid one, as the latter can cause the vanishing gradient problem (meaning an oversaturation of gradients), and this can result in poor training. Moreover, to avoid overfitting in the model, several regularization techniques were implemented, such as: the regularization, dropout and early stopping.The model was then compiled with the Categorical Cross Entropy function and Stochastic Gradient Descent.

Testing the network:

Once the model is fit to the training data, the CNN is evaluated on the training set using the predict() method from the Keras module. Subsequently, the results predicted on the testing set are compared to the test data's actual target values, and a comparative analysis of these results is carried out using performance metrics. Several architectures were built, compared and improved in order to maximize the results of the model. Results are provided in the last section of the report.

Random Forest:

The second machine learning model that was implemented in order to discriminate the different classes of skin lesions is a random forest (RF). RF models are a type of ensemble learning algorithm, which build upon the decision tree algorithm logic (Loupe, 2014). More specifically, random forests are aggregations of small decision trees, which taken in unison form a very powerful classification and prediction model. By taking the majority vote of all the small decision trees, this ensemble algorithm is able to learn and capture the essence of the data.

In order to describe the architecture of RFs, it is important to mention decision trees, as they are the foundations of them. Decision Trees are non-parametric machine learning models which excavate a path for classification through splitting the data based on which features will minimize the entropy and maximize the information gain (Loupe, 2014). Entropy refers to a mathematical measurement which indicates the homogeneity of the data. With each split of the data, the entropy decreases. The information gain measurement is used to calculate how much the entropy will decrease with each split.

RFs have many benefits: though they take time to build, they are substantially faster than other models in their training phase. Moreover, they are able to maintain a high accuracy score, even when some datapoints lack certain features. This is because a singular datapoint is never given an excessive predictive power, thus also eliminating the risk of overfitting the model.

Building a Random Forest:

The random forest built in this project was implemented using Scikit-learn, a library for machine learning and statistical modeling. This package is extremely intuitive, and combines harmoniously with the python language, as the machine learning models are implemented using object-oriented paradigms. Moreover, another advantage of employing this library is its performance metrics module, which enables the user to evaluate the model with great simplicity.

The RF developed for this task analyzes the data through a pandas dataframe, where each column in the CSV file indicates a different feature of all 10015 datapoints. The features used in this classifier are a patient’s age, gender, localization of skin lesion, and type of medical examination performed. Another column in the dataframe corresponds to the target values, which specify the correct class each datapoint belongs to.

The dataset is split into a training set and a test set, with 75% and 25% of the datapoints randomly being allocated to each set respectively. This is common practice in machine learning tasks, as it allows for an accurate portrayal of the model’s learning. Before the forest is built, a search for the best parameters is performed to optimize the accuracy and efficiency of the model without adding unnecessary complexity or hindering its speed.

Training the Random Forest:

The training of the random forest was facilitated by the use of the Scikit-learn library. After importing this module, the forest is initialized by creating a variable which calls the RandomForestClassifier() class. This class can take in some optional parameters, such as a stopping criterion, the maximum depth of a tree, or a minimum number of datapoints for a splitting decision. The criterion used for this project’s implementation is entropy, which is a commonly used information gain metric. Entropy is a measure of dataset purity, which is employed by the random forest algorithm to calculate the “quality” of a split on the dataset in in order to correctly classify the datapoints a meaningful way. In other words, because the entropy value at the root node of a decision tree is 1, the algorithm chooses to perform a decision split based on the amount of entropy minimization. This is also referred to as the information gain; thus, entropy is a measurement to quantify information gain.

The second parameter that was calculated to optimize the random forest is the number of decision trees. The GridSearchCV module, which is found in Scikit-learn’s model_selection library, was used to find this value. This class is fit to our augmented HAM10000 dataset, and scans a dictionary of parameters, returning the optimal number of decision trees for the random forest. Though it has been proven that the accuracy of a random forest does not decline with an increased amount of decision trees, it is favorable to calculate an ideal number of these, to optimize the model’s complexity and speed.

Testing the Random Forest:

Once the model has been fitted to the training data, the RF is evaluated on the training data using Scikit-learn's predict()method. Subsequently, the results predicted on the testing set are compared to the test data's actual target values, and a comparative analysis of these results is carried out using performance metrics.

Results & Evaluation:

It is imperative that the results of a skin cancer classification model be as accurate and precise as possible, as the nature of this task attempts to provide a real-life solution. The advantage of training models for medical purposes is done in an effort to reduce human-error, as well as facilitate the accuracy of results through a plethora of features. Therefore, any margin of error within these systems may have extreme repercussions towards patients.

The way in which these systems were evaluated was through various performance metrics, namely: confusion matrices, accuracy, error rate, precision, recall and F1-score. The confusion matrix is an extremely useful performance metric, which facilitates the visualization of results. It directly compares a classification system's predicted results and target values, which ideally should align within the diagonal of the diagram. The other performance formulas can be derived from a confusion matrix. Accuracy is a statistical metric which is often used to understand the level of performance of a model, at first glance. This formula can be described as the correctly predicted datapoints divided by the sum of all predictions made by the classifier; Inversely, its complementary metric is the error rate, which represents the incorrect number of predictions over the total number of predictions. Therefore, the higher the accuracy, the lower the error rate will be.

The F1-score is a measurement of the balance between precision and recall. Precision score is a measurement to quantify the correctly classified results over all positive result predicted; the recall metric, instead, indicates the percentage of correctly classified positive results over all positive results. The distinction between these two metrics is subtle, yet powerful. For the scope of this project, it is preferred that the precision be maximized over the recall, as cancer classification tasks require extreme rigor and correctness.

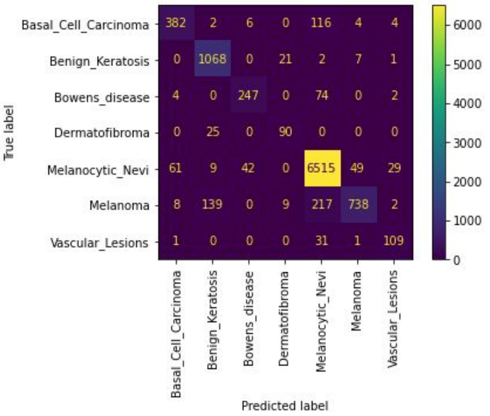
$$F1\ score = 2 * \frac{precision*recall}{precision+recall}$$

Results of the Convolutional Neural Network:

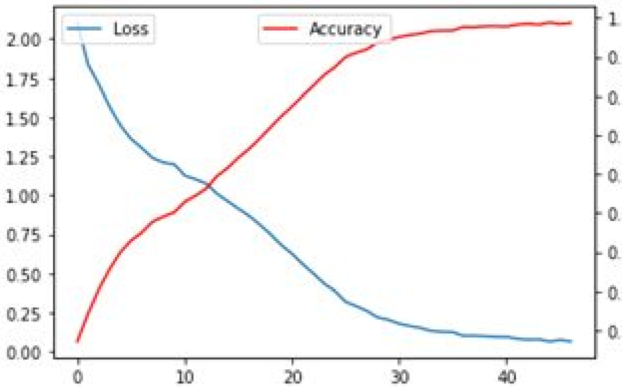
The CNN's results are impressive overall. By analyzing the accuracy score alone, we can observe a radical increment of 24% in performance between the network trained with the original dataset and the augmented one, with accuracy scores of 67% and 91% respectively. This indicates that the artificial data is able to offset the previously biased dataset. In fact, it can be observed in the confusion matrix that the network is able to correctly classify most skin lesions, rather than just the melanocytic nevi one.

Moreover, as the graph below illustrates, the early stopping criterion prevents the model to overfit to the data. In fact, the model was programmed to run for 300 epochs, however the network's training terminated after only 45. This prevents oversaturation of the neurons within the network, which allows the classifier to predict unobserved data correctly and ameliorates its accuracy score.

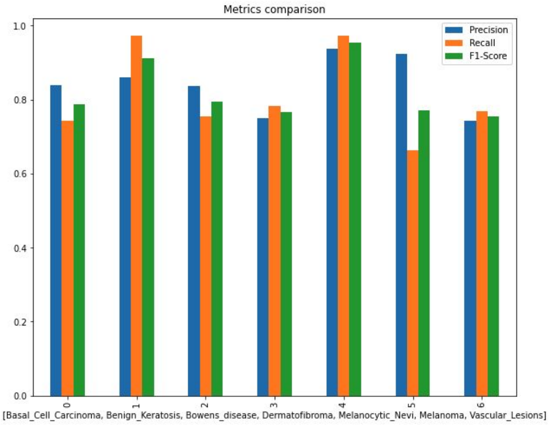
```
Score: 0.9135297054418372
```



True label \ Predicted label	Basal_Cell_Carcinoma	Benign_Keratosis	Bowens_disease	Dermatofibroma	Melanocytic_Nevi	Melanoma	Vascular_Lesions
Basal_Cell_Carcinoma	382	2	6	0	116	4	4
Benign_Keratosis	0	1068	0	21	2	7	1
Bowens_disease	4	0	247	0	74	0	2
Dermatofibroma	0	25	0	90	0	0	0
Melanocytic_Nevi	61	9	42	0	6515	49	29
Melanoma	8	139	0	9	217	738	2
Vascular_Lesions	1	0	0	0	31	1	109

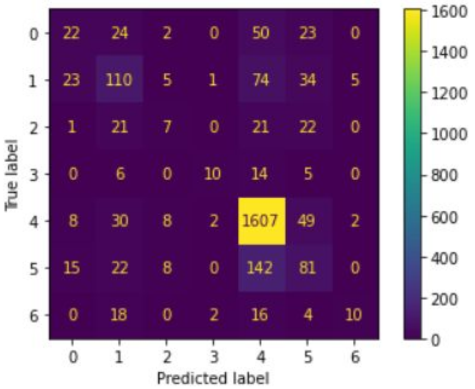


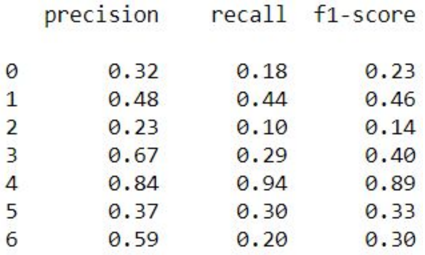
As the testing data is not augmented, the network classifies the Melanocytic Nevi images most accurately due to its large volume of datapoints with respect to other classes. However, the false positives and false negatives metrics are quite low, showing a clear left-to-right diagonal in the matrix. This is further indicated by the F1 score, which suggests the precision and recall values are balanced. The other classes do not provide additional insights. However, as the graph below shows, by comparing the Benign Keratosis and Melanoma classes, which have roughly 1200 datapoints each, interesting disparities are observed. The latter produces a higher number of false negative results, meaning the Melanoma cancer seems to be more difficult for the network to detect. This is not ideal as the scope of this classifier is detecting skin cancers.



Results of the Random Forest Model:

The RF classifier’s results are not optimal, especially when considering the role of this classification task in a real-life scenario. Though the accuracy of the model is 73.8%, it is clear the model has not truly captured the essence of the data, but rather that the melanocytic nevi class is overshadowing all other classes when training, as it has far more datapoints. In fact, this becomes clear while observing the confusion matrix. The metadata for this project was not augmented, as it would be rather fruitless to give thousands of features the same median values, as this in turn would result in a steep artificially produced information gain boundary. As a result, this classifier model does not show quality outcomes. However, these poor results seem to be due to a lack of unbiased data, rather than a fault in the model's architecture. In fact, if there were an evenly distributed amount of data for each class, the RF would be able to produce significant results. This hypothesis may be considered because the precision score for the Melanocytic Nevi class, of 84%, indicates promising results.





```
          precision    recall  f1-score

0              0.32      0.18      0.23
1              0.48      0.44      0.46
2              0.23      0.10      0.14
3              0.67      0.29      0.40
4              0.84      0.94      0.89
5              0.37      0.30      0.33
6              0.59      0.20      0.30
```

Moreover, another factor in the RF's low performance is the limited feature map in the metadata. The model's only inputs are a patient's age, sex, location of skin lesion and method of diagnosis, which are extremely limited and futile factors of the medical analysis of skin cancer. If, for instance, the metadata had presented certain irregularities in the skin lesions or a diameter of skin's abnormalities, these important features would have likely had a powerful impact on the model's classification outcome.

Conclusion:

The aim of this project was to build two systems capable of analyzing and classifying 7 distinct types of skin disease. The data was retrieved from the HAM10000 dataset, which represents the most recently updated medical imaging dataset in the field. The pre-processing of the information available enabled us to create new artificial images, in an attempt to maximize the performance of each classifier and take advantage of both of the model's particular architectures. After partitioning the data into a training and testing set, the models were evaluated using various performance metrics, namely: Accuracy, F1 score, precision, recall and confusion matrices. Overall, the results demonstrate, as anticipated, that CNNs are a more accurate classifier than the RF model. This is due to their feature extraction process, which enables the system to ensure higher performance and generalize better to unseen datapoints.

Though the two classifiers share a common goal, their approach to the solution is extremely different: the CNN attempts to search for patterns and features within each image as a means for classification, whereas the RF analyzes the metadata to elect its decision boundaries. These two models both have their unique advantages and portray different benefits to the same classification task. However, it is clear that one system would be preferred over the other depending on the data available within the medical field. In fact, we argue that the scope of building and comparing different machine learning models is significant as it shows the different ways in which data is "learned" and manipulated. After all, the availability and typology of data often dictates which artificial intelligence system performs best for a singular task. Thus, it may be hypothesized that employing additional medical data, both metadata and images, would significantly improve both models' performance. In fact, future improvements to this project could see an aggregation of multiple medical datasets strung together, in an effort to create a multiclassification system which produces substantial accuracy and uniform results for all skin lesion classes.

References:

Esteva, A., Kuprel, B., Novoa, R., Ko, J., Swetter, S., Blau, H. and Thrun, S., 2017. Dermatologist-level classification of skin cancer with deep neural networks. *Nature*, 542(7639), pp.115-118.

Harvard Dataverse. 2018. [online] Available at: <https://dataverse.harvard.edu/dataset.xhtml?persistentId=doi:10.7910/DVN/DBW86T.>

Loupe, G., 2014. Understanding Random Forests: From Theory to Practice. arxiv.org, [online] 1407(7502), pp.68-105. Available at: <https://arxiv.org/abs/1407.7502> [Accessed 23 October 2020].

Merkulov, A., Yang, C. and Zheng, Y., 2018. Breast cancer screening using convolutional neural network and follow-up digital mammography. SPIE,.

S. Liew, M. Khalil-Hani and R. Bakhteri, "Bounded activation functions for enhanced training stability of deep neural networks on visual pattern recognition problems", Neurocomputing, vol. 216, pp. 718-734, 2016.

Venkatesan, R. and Li, B., 2017. Convolutional Neural Networks In Visual Computing. 1st ed. Boca Raton: CRC Press, pp.89-114, 134-141.

Y. LeCun, L. Bottou, Y. Bengio and P. Haffner, "Gradient-based learning applied to document recognition", Proceedings of the IEEE, vol. 86, no. 11, pp. 2278-2324, 1998.