ISYE 6740 – Summer 2023

Final Project

Early Detection of Skin Diseases using Images

Muhammad Ahmad, Luke Williams

Introduction:

Skin diseases, including skin cancers, pose a substantial global health concern, affecting millions of people worldwide. Early detection and accurate diagnosis are key for successful treatment and improving patient outcomes. To address this critical issue, our project aims at developing a predictive model for the early detection of skin diseases based on dermoscopic and dermatological images.

Early detection of skin diseases is crucial for several reasons, the first being improved diagnosis. In cases of skin cancer, detecting malignancies at an early stage significantly improves the chances of successful treatment and patient survival. Early detection can also prevent the disease from spreading and from advancing to a more critical stage, which is much more difficult to treat in later stages. Early detection also allows for less invasive and less costly treatment options compared to advanced-stage diseases, reducing the economic burden on patients. Early diagnosis can prevent the progression of certain skin diseases, leading to improved quality of life for patients by minimizing discomfort and complications. Most importantly, skin cancers and other serious skin diseases may lead to death. Early detection and treatment can reduce the chances of death in the patients.

In this project we will be looking at dermoscopic and dermatologic images. Dermoscopy is a non-invasive imaging technique that allows clinicians to examine skin lesions at a higher magnification than normal. It provides valuable information about the lesion's structure, pigmentation, and vascular patterns, which aids in the identification of malignant (cancerous) features. These images are an essential resource for dermatologists to make accurate diagnoses. Likewise, dermatologic images, which include clinical photographs and histopathological images, also have similar functionality to dermoscopic images. They provide additional context for dermatologists to assess the lesion's appearance, size, and location, which help make evaluations of skin diseases.

In conclusion, this project aims to harness the power of machine learning and image analysis to develop an early detection model for skin diseases. By leveraging dermoscopic and dermatologic images, we will be able to provide clinicians with a powerful tool to enhance diagnosis, treatment, and patient care in the practice of dermatology.

Objectives:

The primary objectives of this project revolve around leveraging advanced technology and image analysis to capture early detection of skin diseases. By using dermoscopic and dermatologic images, this project aims to develop different sophisticated predictive models that can effectively distinguish between different skin diseases and disorders. This capability will be particularly valuable in cases of skin cancer, where early diagnosis significantly impacts treatment success and will lead to increased patient survival rates.

To ensure the model's effectiveness and reliability, we will evaluate the different models on the test data using metrics such as sensitivity, accuracy, and the confusion matrices. The evaluation process will help validate the model's performance and ensure that it can generalize well to new cases which is critical for practical application in real-world clinical settings.

Overall, this project represents a significant step forward in the field of dermatology, where cutting-edge technology is harnessed to support medical professionals in making accurate and fast diagnoses. By combining machine learning algorithms, advanced image processing, and a diverse dataset of dermoscopic and dermatologic images, the models have the potential to improve early detection and management of skin diseases, ultimately benefiting patients in both the short and long term.

Data:

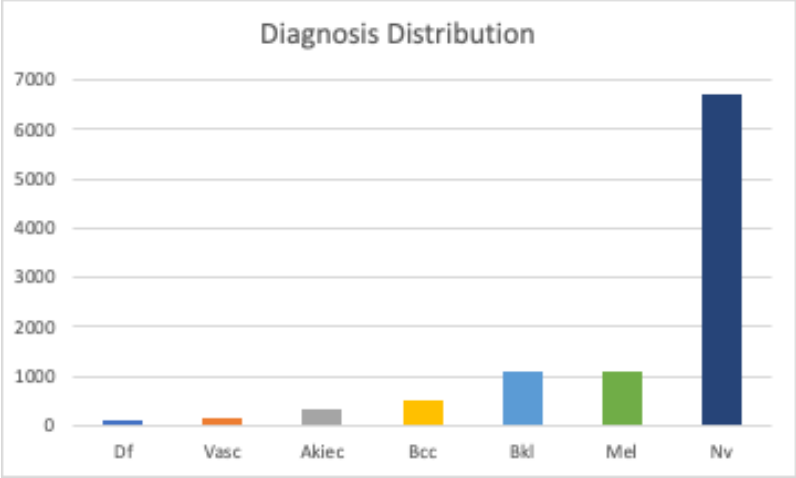
We plan to utilize two datasets for this project, the Dermnet dataset from Kaggle and the HAM10000 dataset from Harvard Dataverse. The Dermnet dataset contains a collection of dermatologic images covering 23 different skin diseases and disorders. Of the 23 diseases and disorders we only looked at ones that had at least 450 images so we could have a large enough population to have both validation and test datasets. The HAM10000 dataset consists of dermoscopic images specifically focused on skin cancer cases. The combination of these datasets will provide a diverse set of images for training and testing our model.

The Dermnet dataset consists of 19,500 images representing various skin disorders and diseases, with each image having a different resolution. The dataset has a total of twenty-three common skin conditions. However, for our project, we have selected thirteen specific diseases from this dataset to focus on during our research and model development. By narrowing down our scope to these thirteen diseases, we aim to create a more targeted and specialized predictive model for accurate early detection and diagnosis. The criteria we chose was to filter any category with less than 450 training images because that would be too little data to have a good model, which could create problems detecting the disease. Therefore, we decided to filter them, the 13 diseases are:

1. Acne and Rosacea
2. Actinic Keratosis Basal Cell Carcinoma and other Malignant Lesions
3. Atopic Dermatitis
4. Eczema
5. Light Diseases and Disorders of Pigmentation
6. Melanoma Skin Cancer Nevi and Moles
7. Nail Fungus and other Nail Disease
8. Psoriasis Lichen Planus and related diseases
9. Seborrheic Keratoses and other Benign Tumors
10. Systemic Disease

11. Tinea Ringworm Candidiasis and other Fungal Infections
12. Vascular Tumors,
13. Warts Molluscum and other Viral Infections

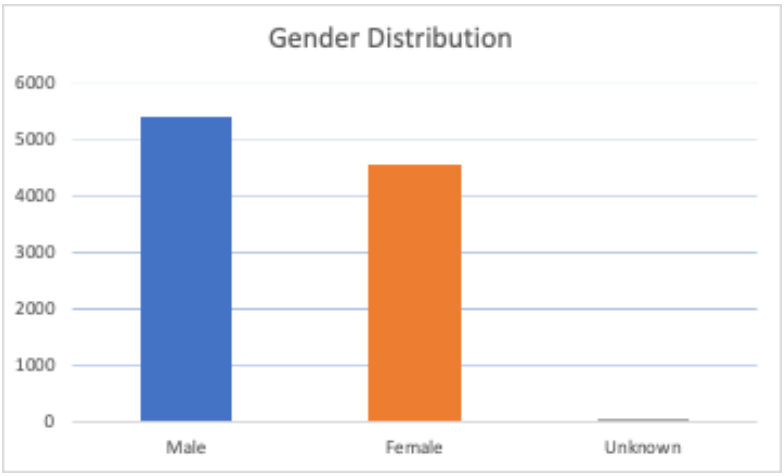
The HAM10,000 dataset comprises a collection of 10,000 dermatoscopic images. In addition to the images, this dataset provides valuable supplementary information, including age, sex, and localization of the skin lesions. This comprehensive dataset offers a wealth of data that can be leveraged to develop a more comprehensive and holistic predictive model for early detection and diagnosis of skin diseases. The diagnosis count is distributed as follows:



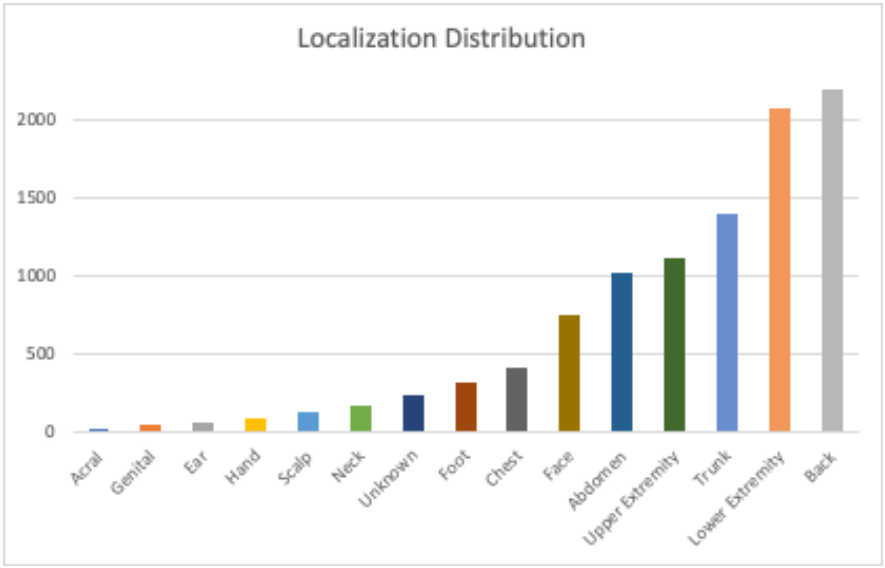
There are seven different diagnoses in the HAM10000 dataset:

1. Df: Dermatofibroma
    a. Df is often asymptomatic, which makes it hard for patients to detect, but it can lead to constant itching and pain if not treated.
2. Vasc: Vascular lesions
    a. Vascular lesions are more commonly known as birthmarks. These are common abnormalities in the skin and tissue.
3. Akiec: Actinic Keratose / Bowen's disease
    a. This is a very early form of skin cancer that's very easily treatable. It's characterized by a red scaly patch of skin.
4. Bcc: Basal cell Carcinoma
    a. This is a type of skin cancer that develops mostly on areas of skin exposed to the sun (mainly the face).
5. Bkl: Benign karatosis like lesions
    a. This is the most common type of benign skin lessions. They often grow slowly and will often cause sharp pain in the patients.
6. Mel: Melanoma
    a. Melanoma is a malignant cancer that form melanocytes, which are cells that color the skin). Excessive exposure to sunlight and unusual moles can increase the risk of melanoma.
7. Nv: Melanocytiv Nevi
    a. Melanocytiv nevi are common lesions are moles and are benign and often don't require any treatment, although a small percentage develop melanoma in them.

In the dataset, besides the diagnosis, there are various other features, including age. Notably, there were 57 instances with missing ages, which were handled by taking the median age as a substitute. This data imputation approach helps to maintain the dataset's integrity and ensures that the predictive model can utilize the available age information effectively in the early detection and diagnosis of skin diseases. By addressing missing data and incorporating relevant features like age, the model becomes more capable of providing valuable insights for patient care. The dataset also includes the sex of the patients. The distribution of sex is shown below:



The last feature in the dataset is the localization (area of the body), which is especially important for characterizing skin diseases accurately. This feature provides critical information about the specific location on the body where the skin lesions are observed. The localization data can add valuable insights into how different skin diseases manifest in various regions of the body, contributing to a more precise diagnosis for the model. The localization distribution is shown below:



Methodology:

Methodology for Dermnet Dataset:

We used Scikit-learn to split the training dataset into 93% for training and 7% for validation randomly and used stratify approach so that we didn't gather based on distributions (i.e., picking more from some categories). There was a separate test dataset as mentioned before with data for every categories. Then we used tensorflow ImageDataGenerator function to augment data based on randomly changing our images by using transformation such as rotation, width change, height changes and horizontal flip and zoom.

Models:

We started off with a simple convolutional neural network (CNN) model using Keras, a high-level API for TensorFlow. The model contains three convolutional layers followed by a max pooling layer. After the convolutional and pooling layers, the model has a flattened layer, a fully connected layer (dense layer) with 512 neurons and ReLU activation function, and finally a softmax layer for multi-class classification. The number of neurons in the softmax layer corresponds to the number of classes in our dataset.

After defining the model, it is compiled with the Adam optimizer and categorical crossentropy as the loss function, which is appropriate for multi-class classification tasks. The model's performance is measured using accuracy. However, the model didn't bear any fruits, we were only able to achieve validation accuracy around 25% even after 18 epochs and it only had very acute marginal gains. Hence, we decided to move on to next model.

Our next approach was to use the InceptionV3 neural network model pre-trained on ImageNet from TensorFlow. It was loaded without the top classification layer. It acts as the base model for feature extraction. Our images' dimensions and colour channels define their input shape. A GlobalAveragePooling2D layer follows the base model, reducing the spatial dimensions and managing the number of parameters. It performs an average pooling operation on each feature map of the tensor output from the base model. Next, we have a fully connected Dense layer with 1024 neurons and ReLU activation function. This layer allows our model to learn more complex patterns based on the features extracted previously. The final Dense layer has a neuron for each class in our dataset and uses the softmax activation function to output the class probabilities. This layer serves as the classifier in our model. The final model is defined with the same inputs as the base model and outputs from the classifier layer. After defining the model, it is compiled with the Adam optimizer and the categorical cross-entropy loss function. We tried exploring this model by adding regularization penalties for overfitting. Unfortunately, this also did not give us desirable accuracy. We tested a few times by changing some parameters in the model, but it did not seem to converge, there was too much overfitting, and our validation test score was quite low.

Our best results were by a convolutional neural network (CNN) model for a multiclass image classification task by using a pre-trained Xception model (again we Used tensor flow for it). The Xception model, trained on the ImageNet dataset, is used as the base model for feature extraction. The input shape is defined according to our image dimensions and color channels. Following the base model, a GlobalAveragePooling2D layer reduces the spatial dimensions of the model's output. This process condenses each feature map in the tensor to a single average value, reducing the number of parameters

and minimizing the risk of overfitting. A Dense layer with 1024 neurons and a ReLU activation function follows. This fully connected layer facilitates the learning of complex patterns based on the features extracted by the base model. It also includes L2 regularization, which helps prevent overfitting by adding a penalty to the loss function for large weights. We tried using different L2 regularization, our best results were with a value of 0.01. The final Dense layer is a classifier with a neuron for each class in our dataset and uses a softmax activation function to produce class probabilities. The model uses the same inputs as the base model and the outputs from the classifier layer. After defining the model, it is compiled with the Adam optimizer and the categorical crossentropy loss function, suitable for multiclass classification. We achieved ~65% (64.649%) test accuracy with this. We will discuss the results in depth in the results section.

Lastly, we explored other machine learning model to see if simpler techniques work better. For this we used we leveraged a pre-trained ResNet50 model to extract high-level features from a dermatological image dataset. We froze the layers of the pre-trained model to maintain its learned features and used the model as a feature extractor for our own classification task. We then transformed our train, validation, and test datasets into these high-level features. This is an example of transfer learning, where we utilized a model trained on a large dataset (ImageNet) and adapted it to our specific task (classifying dermatological diseases), aiming to boost performance by utilizing the learned representations. Then we applied Principal Component Analysis (PCA), a dimensionality reduction technique, to the extracted features. First, we flatten the 2D feature maps into 1D vectors, resulting in an array of shapes (num_samples, 6 * 6 * 2048) for each of the train, validation, and test datasets. After reshaping, we applied PCA, retaining 200 principal components. We could have used a much lower principal component number and we tested it with n=10 and n=50 but there was not much improvement, and we were not concerned about compute power too much, so we used 200 anyways). The goal of PCA is to reduce the dimensionality of the data while preserving as much variance as possible. By doing this, we simplified the data without losing too much information and made subsequent computations more efficient. The PCA model is fitted only on the training dataset to avoid data leakage, and then transformed using both the validation and test datasets. After that, we used lazypredict library for trying out different ML models. None of them performed well, the highest validation accuracy we got was 27% with XGBoost Classifier so we did not bother exploring it further. We will have the results for this in appendix.

Methodology for Skin Cancer Dermoscopic Dataset:

We used Scikit-learn (train_test_split function) to split the dataset into 8% for testing and 7.6% for validation randomly and rest of the data for training. The stratified approach was used so that we do not gather based on distributions (i.e., picking more from some categories or none from some). Then we used the TensorFlow ImageDataGenerator function to augment data based on randomly changing our images by using transformations such as rotation, width, height, horizontal flip and zoom. This allows more validation and test accuracy and less overfitting.

Models:

We started off with using a pre-trained Xception model, a convolutional neural network (CNN) model for a multiclass image classification task. This model is similar to the one we used for Dermnet data. Following the base model, a Flatten layer is used to reduce the spatial dimensions of the output

tensor, converting it into a single long vector. A Dense layer with 1024 neurons and a ReLU activation function follows. It also includes L2 regularization, which helps prevent overfitting by adding a penalty to the loss function for large weights. We ended up using L2 regularization of 0.01. We tried without regularization and smaller value, but the model gave higher overfitting i.e., low validation accuracy. The final Dense layer is a classifier with a neuron for each class in our dataset and uses a softmax activation function to produce class probabilities.

The model uses the same inputs as the base model and the outputs from the classifier layer. After defining the model, it is compiled with the Adam optimizer and the categorical crossentropy loss function, which is used for multiclass classification. The Adam optimizer is an algorithm for gradient-based optimization of stochastic objective functions. It is known for its efficiency and effectiveness in machine learning tasks, particularly in complex neural networks like this one. The Adam optimizer works by adaptively changing the learning rates for each weight in the model. This means that it calculates individual learning rates for different parameters, which can be especially useful when dealing with high-dimensional datasets or complex neural networks. We used following parameters:

- lr=0.001: This is the learning rate, determining the step size during optimization.
- beta_1=0.9 and beta_2=0.999: These values control how the optimizer uses gradient information from past steps to influence the current step.
- epsilon=1e-07: This small constant is for numerical stability, to avoid division by zero.
- decay=0.0: This is the learning rate decay, indicating how the learning rate decreases over time, which is not used in this case.
- amsgrad=False: This parameter indicates we are using the standard Adam algorithm, not its variant AMSGrad.

We were able to achieve 76.75% validation accuracy and 72.44 test accuracy. (Note: Tables related to this will be in appendix section)

For this dataset, our best model was a pre-trained Densenet21 model, a convolutional neural network (CNN) model for a multiclass image classification task. The DenseNet121 model, trained on the ImageNet dataset, is used as the base model for feature extraction. The input shape is set to match the dimensions of our images. Some layers in the base model are frozen (set untrainable) to retain their pre-trained weights. The output from the base model is passed to a GlobalAveragePooling2D layer to reduce spatial dimensions, then a Dense layer with 256 neurons, a ReLU activation function, and L2 regularization (0.01). It is followed by a BatchNormalization layer and a final Dense layer with 7 neurons (one for each class) using softmax activation.

The model uses an Adam optimizer with a learning rate that decays exponentially over time. The learning rate starts from 0.0005, and after each 10000 steps, it decays by a rate of 0.9. Other parameters in the optimizer include beta_1=0.9, beta_2=0.999, and epsilon=1e-08, all of which are standard values in the Adam algorithm.

We were able to achieve overall 85.17% validation accuracy and 84.16% accuracy on test data with this model.

Evaluation and Final Results:

The Dermnet Dataset Results:

We were able to achieve highest prediction accuracy with Xception CNN model. The results were as follows:

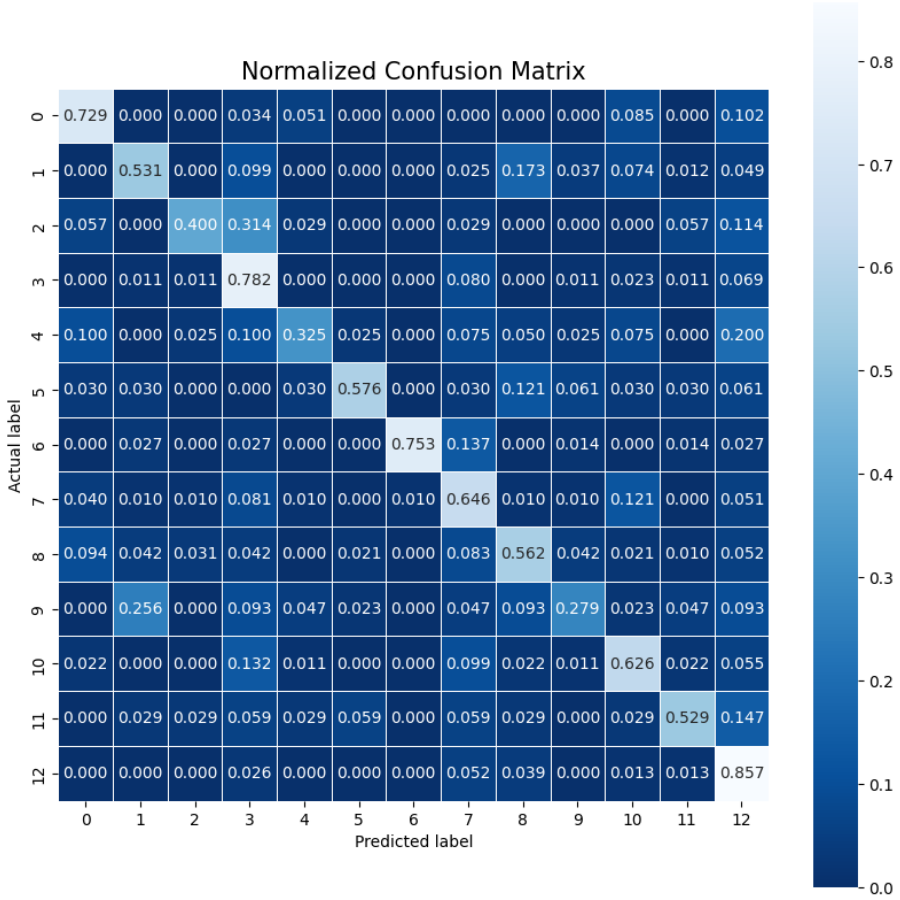
```
Epoch 18/20 49/49 [==============================] - 120s 2s/step - loss: 0.2039
- accuracy: 0.9445 - val_loss: 1.5925 - val_accuracy: 0.6563

Found 3117 images belonging to 13 classes. 13/13 [==============================]
- 10s 733ms/step - loss: 1.6135 - accuracy: 0.6465 Test accuracy:
0.6464956998825073
```

This shows validation accuracy of 65.63% and 64.65% accuracy on test dataset. We ran more epochs but stored the model with highest validation accuracy.

This is a classification report obtained:

	precision	recall	f1-score	support
Acne and Rosacea Photos	0.66	0.73	0.69	59
Actinic Keratosis Basal Cell Carcinoma and other Malignant Lesions	0.67	0.53	0.59	81
Atopic Dermatitis Photos	0.67	0.40	0.50	35
Eczema Photos	0.54	0.78	0.64	87
Light Diseases and Disorders of Pigmentation	0.57	0.33	0.41	40
Melanoma Skin Cancer Nevi and Moles	0.76	0.58	0.66	33
Nail Fungus and other Nail Disease	0.98	0.75	0.85	73
Psoriasis pictures Lichen Planus and related diseases	0.57	0.65	0.60	99
Seborrheic Keratoses and other Benign Tumors	0.64	0.56	0.60	96
Systemic Disease	0.46	0.28	0.35	43
Tinea Ringworm Candidiasis and other Fungal Infections	0.63	0.63	0.63	91
Vascular Tumors	0.60	0.53	0.56	34
Warts Molluscum and other Viral Infections	0.54	0.86	0.66	77
accuracy			0.62	848
macro avg	0.64	0.58	0.60	848
weighted avg	0.64	0.62	0.61	848

Normalized Confusion Matrix:



Keys for confusion matrix:

0: Acne and Rosacea

1: Actinic Keratosis Basal Cell Carcinoma and other Malignant Lesions

2: Atopic Dermatitis

3: Eczema

4: Light Diseases and Disorders of Pigmentation

5: Melanoma Skin Cancer Nevi and Moles

6: Nail Fungus and other Nail Disease

7: Psoriasis pictures Lichen Planus and related diseases

8: Seborrheic Keratoses and other Benign Tumors

9: Systemic Disease

10: Tinea Ringworm Candidiasis and other Fungal Infections

11: Vascular Tumors

12: Warts Molluscum and other Viral Infections

The model trained using the Xception architecture and applied to a dataset of dermatological conditions achieved a significant level of performance, as demonstrated by the evaluation metrics. In the best epoch during training, the model reached an impressive accuracy of 94.45%. This indicates a high level of proficiency in correctly classifying the training data, demonstrating that the model has effectively learned from the provided examples.

However, the model's ability to generalize to unseen data was lower, as observed from the validation and test set results. The highest validation accuracy across all epochs was 65.63%, indicating that the model's performance dropped when evaluated on the validation set. The goal is to minimize this performance gap to ensure the model is not overfitting and is able to generalize well.

The test accuracy was slightly lower at 64.65%, which is a more realistic indication of the model's performance on entirely new data. The evaluation on the test set further confirmed the model's ability to classify the different dermatological conditions correctly about two-thirds of the time.

In the provided classification report, the model exhibited varying performance across different classes. Some conditions had higher F1-scores, implying a good balance between precision and recall for those classes. However, others had lower scores, indicating potential difficulties in distinguishing these conditions. We can also there is somewhat high correlation between some of the categories for example between Systemic Disease (9) and Actinic Keratosis Basal Cell Carcinoma and other Malignant Lesions, we have prediction correlation of 0.256.

In conclusion, while the model demonstrates a considerable ability to classify skin conditions based on images, there is still room for improvement. Fine-tuning the model and increasing the amount of data for underrepresented classes could be potential steps towards improving the model's performance. I think adding normal skin data would be a good control as well. We personally think with slightly more data the same ML algorithms will perform far better.

The Skin Cancer Dermoscopic Dataset Results:

The DenseNet121 model with 26 epochs performed exceptionally well on the skin cancer classification task using Dermascopic images.

```
Epoch 26/30 42/42 [==============================] - 39s 935ms/step - loss:
0.0931 - accuracy: 0.9787 - val_loss: 1.0552 - val_accuracy: 0.8517
```

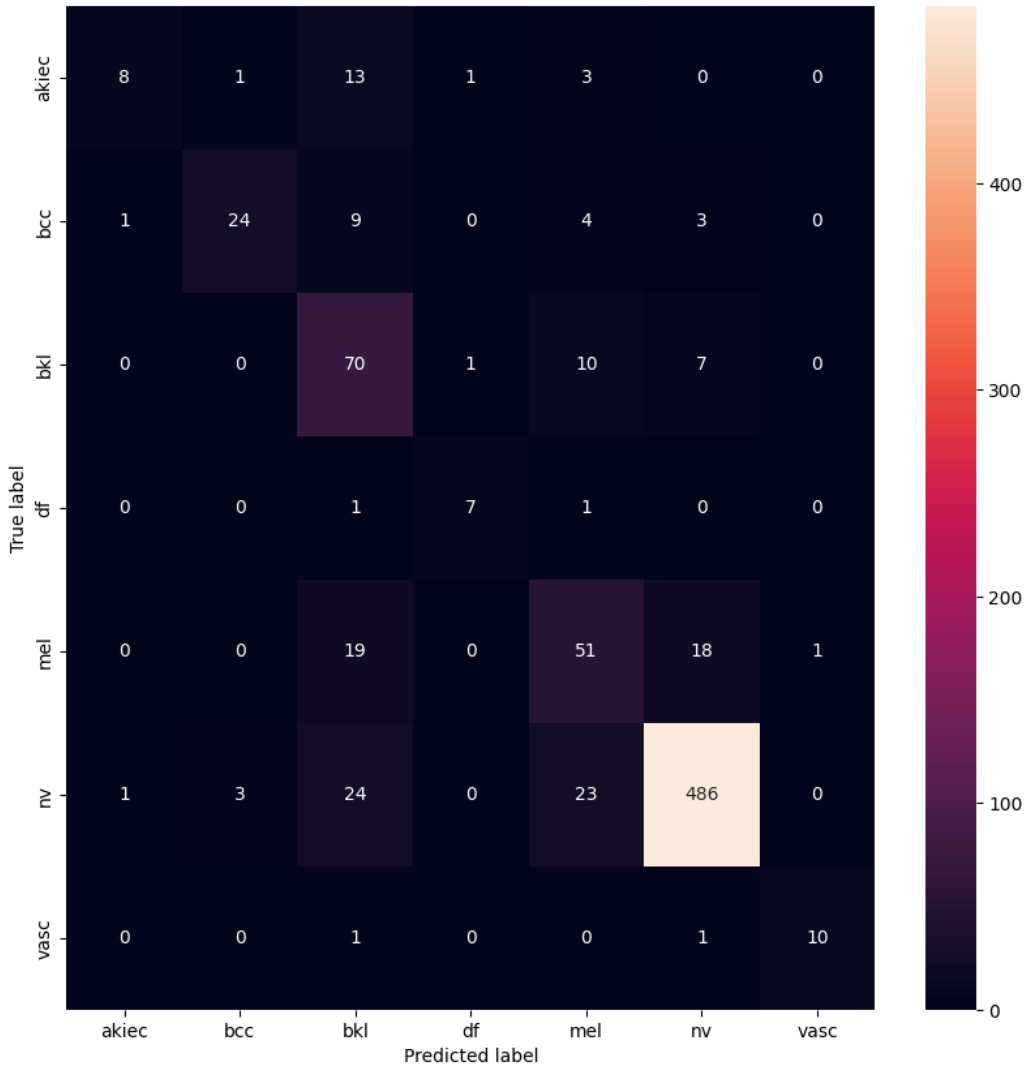
```
26/26 [==============================] - 2s 18ms/step - loss: 1.0402 - accuracy:
0.8416 Loss: 1.0401535034179688 Accuracy: 0.8416458964347839
```

This shows validation accuracy of 85.17% and 84.17% accuracy on test dataset. We ran more epochs but stored the model with highest validation accuracy.

Here is a classification report:

	precision	recall	f1-score	support
akiec	0.80	0.31	0.44	26
bcc	0.86	0.59	0.70	41
bkl	0.51	0.80	0.62	88
df	0.78	0.78	0.78	9
mel	0.55	0.57	0.56	89
nv	0.94	0.91	0.92	537
vasc	0.91	0.83	0.87	12
accuracy			0.82	802
macro avg	0.76	0.68	0.70	802
weighted avg	0.84	0.82	0.82	802

Confusion Matrix:



The overall accuracy on the test set reached around 84.16%. This indicates that, out of every 100 images, the model correctly identified the class of approximately 84 images, which is a solid performance. The model's loss on the test set, a measure of error, was approximately 1.04.

Looking at the class-wise performance in the precision-recall report, the model showed varying effectiveness for different classes. The 'nv' class, with 537 instances in the test set, had the highest precision (0.94), recall (0.91), and F1-score (0.92) among all the classes. This high performance might be due to the larger number of 'nv' instances providing more information for the model to learn from. On the other hand, the 'akiec' class had the lowest precision, recall, and F1-score, indicating that the model had a harder time classifying this class. One potential reason could be the comparatively fewer instances of 'akiec' in the training set.

It's also worth noting that the model seemed to generalize well from the training to the validation set, as seen by similar accuracy values. However, the validation loss is higher than the training loss, indicating that the model might be overfitting to the training data to some extent. It's performing well, but there might be room for improvement through techniques such as more regularization, different architecture choices, or more data. In conclusion, the DenseNet121 model provides a strong baseline for the skin lesion classification task. Its performance could potentially be enhanced with further tuning and experimentation.

Conclusion:

To conclude, the results of this project demonstrate significant progress towards the early detection of skin diseases using dermoscopic and dermatologic images. We can get correct classification for Dermnet images data or dermalogic images (which you can even take from your phone) roughly 2 out of 3 times. To help increase the accuracy we believe we need to further fine tune our models and collect more data. We achieved great success on the dermascopic cancer dataset, obtaining overall 84% accuracy on the test data. This outcome holds significant promise for enhancing cancer diagnosis and treatment.

Despite the accuracy of the model, it's important to acknowledge the challenges in accurately diagnosing various skin diseases. The variations in lesion appearance and the diversity of skin conditions present difficult obstacles, especially for a machine learning model. Continuous research, collaboration with dermatology experts, and the integration of more diverse datasets are essential to improve the model and achieve higher accuracy across all skin diseases.

Overall, this project creates a foundation for the early detection of skin diseases, offering valuable insights and paving the way for future research and development in this critical area of healthcare. The classification of skin lesions and distinguishing malignant cases are promising steps towards improved patient outcomes, and the project's outcomes have the potential to revolutionize the way skin diseases are diagnosed. The combination of expert healthcare opinions and techniques along with the development of machine learning models will be transformative in the future diagnosis success rate, which will hopefully lead to increase survival rates for the patients.

References:

Dermnet: https://www.kaggle.com/datasets/shubhamgoel27/dermnet

HAM10000: https://dataverse.harvard.edu/dataset.xhtml?persistentId=doi:10.7910/DVN/DBW86T

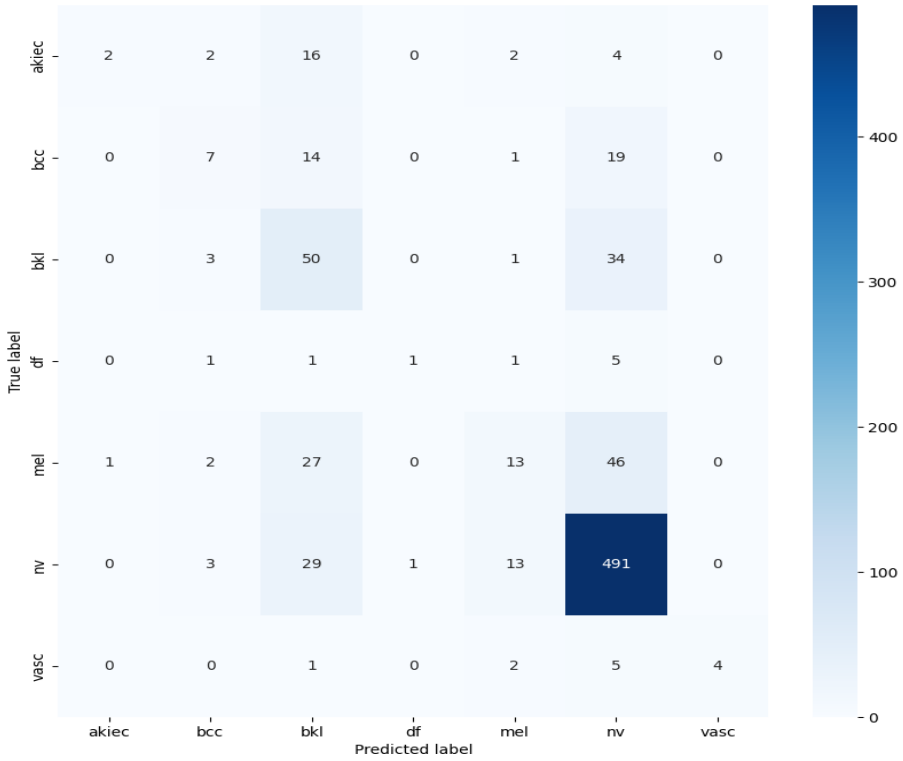
Appendix:

Xception model results for The Skin Cancer Dermoscopic Dataset:

Classification report:

	precision	recall	f1-score	support
akiec	0.67	0.08	0.14	26
bcc	0.39	0.17	0.24	41
bkl	0.36	0.57	0.44	88
df	0.50	0.11	0.18	9
mel	0.39	0.15	0.21	89
nv	0.81	0.91	0.86	537
vasc	1.00	0.33	0.50	12
accuracy			0.71	802
macro avg	0.59	0.33	0.37	802
weighted avg	0.69	0.71	0.67	802

Confusion Matrix:



Results from Simple ML models using lazypredict for Dermnet dataset:

```
100%|██████████| 29/29 [15:23<00:00, 31.86s/it]
```

Model	Accuracy	Balanced Accuracy	ROC AUC	F1 Score
XGBClassifier	0.32	0.27	None	0.30
LGBMClassifier	0.31	0.27	None	0.30
NearestCentroid	0.25	0.25	None	0.25
RandomForestClassifier	0.30	0.24	None	0.27
ExtraTreesClassifier	0.30	0.24	None	0.27
QuadraticDiscriminantAnalysis	0.26	0.24	None	0.25
SVC	0.29	0.23	None	0.26
LogisticRegression	0.26	0.23	None	0.25
LinearDiscriminantAnalysis	0.26	0.22	None	0.25
LinearSVC	0.26	0.22	None	0.24
BernoulliNB	0.24	0.22	None	0.23
KNeighborsClassifier	0.23	0.22	None	0.23
CalibratedClassifierCV	0.26	0.21	None	0.23
RidgeClassifier	0.26	0.20	None	0.22
RidgeClassifierCV	0.26	0.20	None	0.22
BaggingClassifier	0.22	0.20	None	0.21
SGDClassifier	0.20	0.18	None	0.20
DecisionTreeClassifier	0.19	0.18	None	0.19
AdaBoostClassifier	0.21	0.17	None	0.19
Perceptron	0.19	0.16	None	0.19
ExtraTreeClassifier	0.16	0.15	None	0.16
PassiveAggressiveClassifier	0.16	0.14	None	0.16
GaussianNB	0.13	0.14	None	0.11
LabelPropagation	0.11	0.12	None	0.09
LabelSpreading	0.11	0.12	None	0.09
DummyClassifier	0.12	0.08	None	0.02

Model	Time Taken
XGBClassifier	293.73
LGBMClassifier	28.12
NearestCentroid	0.13
RandomForestClassifier	19.44
ExtraTreesClassifier	4.68
QuadraticDiscriminantAnalysis	0.77
SVC	31.80
LogisticRegression	0.92
LinearDiscriminantAnalysis	0.55
LinearSVC	104.00
BernoulliNB	0.15
KNeighborsClassifier	0.23
CalibratedClassifierCV	373.34
RidgeClassifier	0.14
RidgeClassifierCV	0.44
BaggingClassifier	24.26
SGDClassifier	6.18
DecisionTreeClassifier	3.62
AdaBoostClassifier	17.37
Perceptron	0.72
ExtraTreeClassifier	0.14
PassiveAggressiveClassifier	1.16
GaussianNB	0.14
LabelPropagation	3.77
LabelSpreading	7.82
DummyClassifier	0.08