Overview

I will use the Skin Cancer dataset to predict whether the image is a benign or malignant tumor.The challenge2020 dataset resized to 512x512 is available here on kaggle SIIM-ISIC Melanoma Classification is a dataset used in machine learning to predict whether an image of a skin lesion is melanoma or not. The dataset contains a total of 33,126 images of skin lesions, where 32,312 are benign and 814 are malignant.

Dataset - https://challenge2020.isic-archive.com/

Goals

1. Collect data and conduct data analysis. Visual reports.

2. Evaluate Classifica.on Models (Decision Trees, Random Forests, Linear Regression) with Deep Learning Models (CNN, RNN etc) and based on the outcome Design and implement one or more deep learning systems, experiment with various algorithms to maximize the learning capability. Evaluate the performance and document findings.

3. Cost functions should be carefully thought through and justified.

4. Image segmentation or object detection can be carried out to extract the pathological regions for refined detection and analysis.

5. Evaluate if addition or integration of demographic information such as age, ethnic groups, gender etc. may give better predictions.

6. Explore unsupervised learning on large unlabelled data.

7. Fine tuning on a limited source of pathological data.

Dataset Overview

Images are also provided in JPEG resized a uniform 512x512..

Metadata is also provided outside of the JPEG format, in CSV files. See the Columns section for a description.

We have to predict a binary target for each image. The model model should predict the probability (floating point) between 0.0 and 1.0 that the lesion in the image is malignant (the target). In the training data, train.csv, the value 0 denotes benign, and 1 indicates malignant.

Metadata:

- **image_name** - unique identifier, points to filename of related DICOM image
- **patient_id** - unique patient identifier
- **sex** - the sex of the patient (when unknown, will be blank)
- **age_approx** - approximate patient age at time of imaging
- **anatom_site_general_challenge** - location of imaged site
- **diagnosis** - detailed diagnosis information (train only)
- **benign_malignant** - indicator of malignancy of imaged lesion
- **target** - binarized version of the target variable

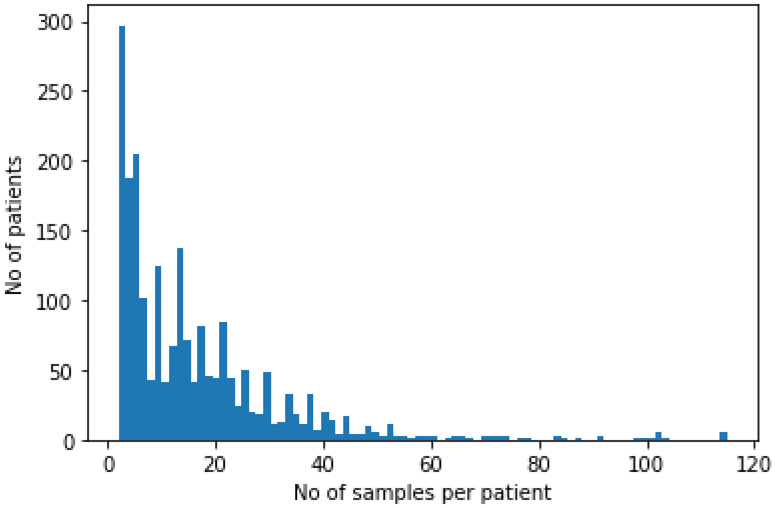
Exploratory Data Analysis

Before coming to Machine learning I have to understand what type of data we are dealing with and how it is correlated with itself.

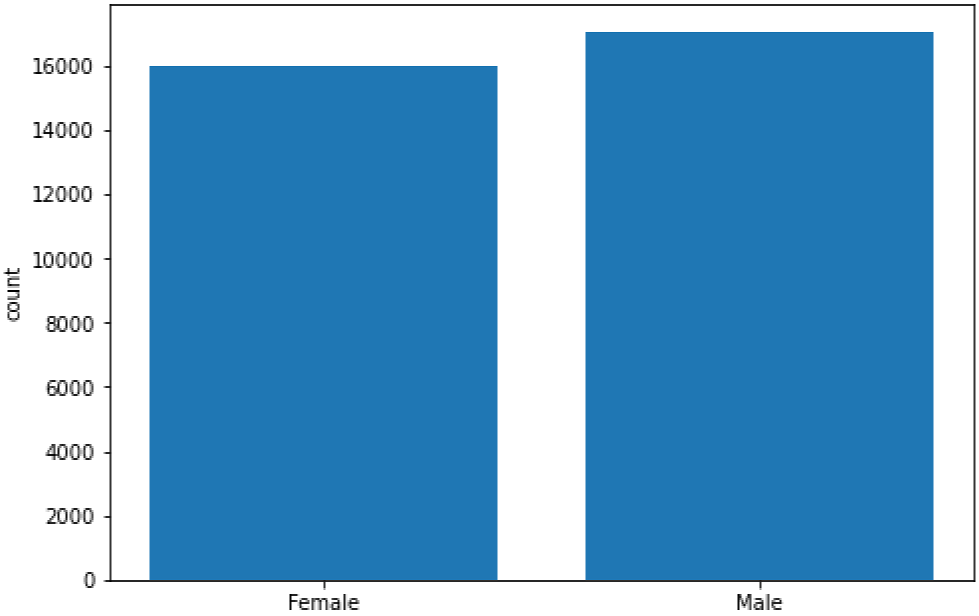
No of samples taken from patients frequency

Observing the number of patients and no of total samples, I came to the following insights.

- All the patients gave at least **2 samples.**
- Maximum no of samples taken from a single patient is **115.**
- On an average each patient gave **16 samples**
- Median of samples of image per patient is **12**
- Mode of samples of image per patient is **3**

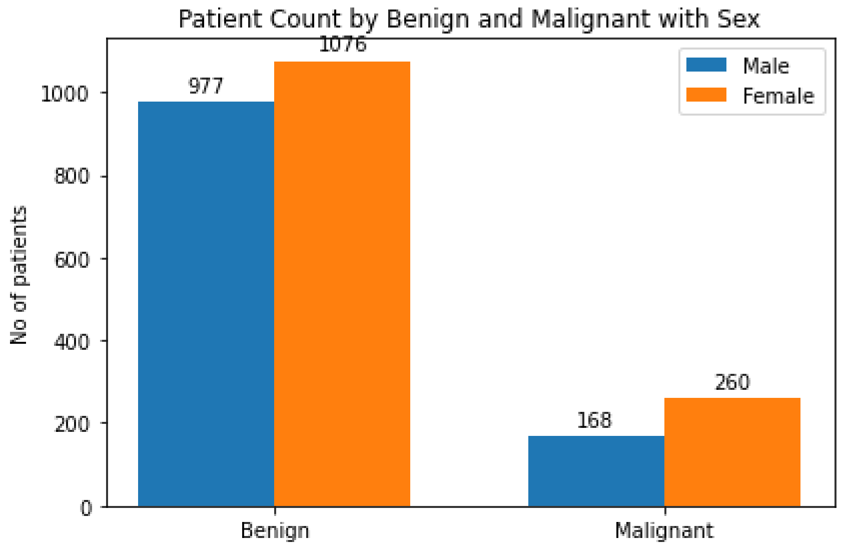


Age Distribution



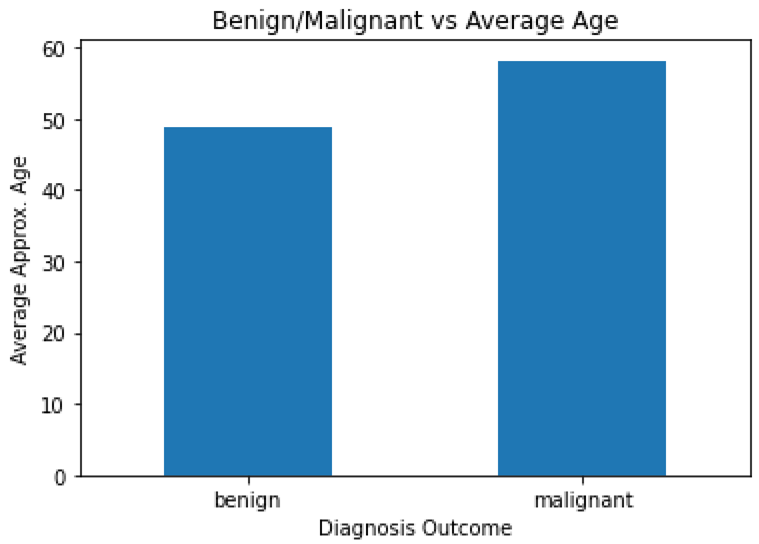
The distribution of patients is almost equal the sample dataset has a balanced distribution of patients based on their gender

Distribution of Patients by Gender

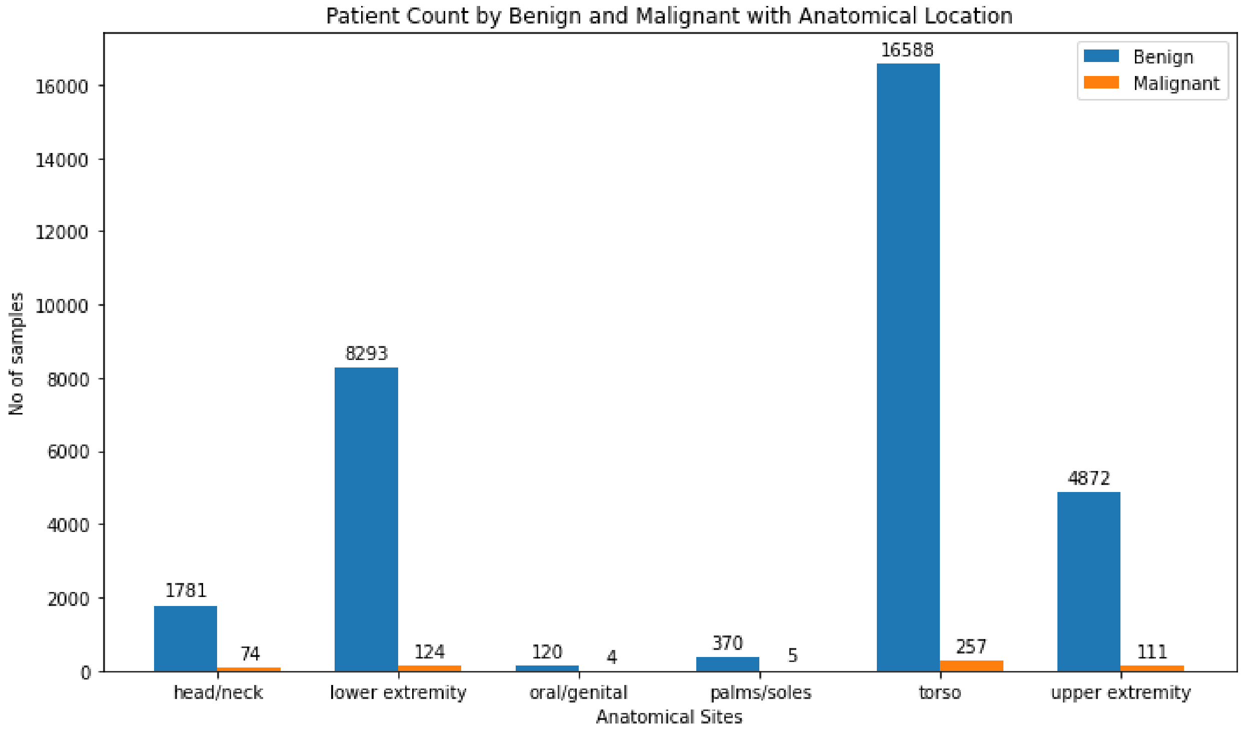


Here we observe that among the unique patients providing samples,

- **Melanoma** is more prevalent in Women
- Among the Male patients, almost **24%** are at malignant stage
- On the other hand, among Female patients, about **17%** are at **malignant stage**



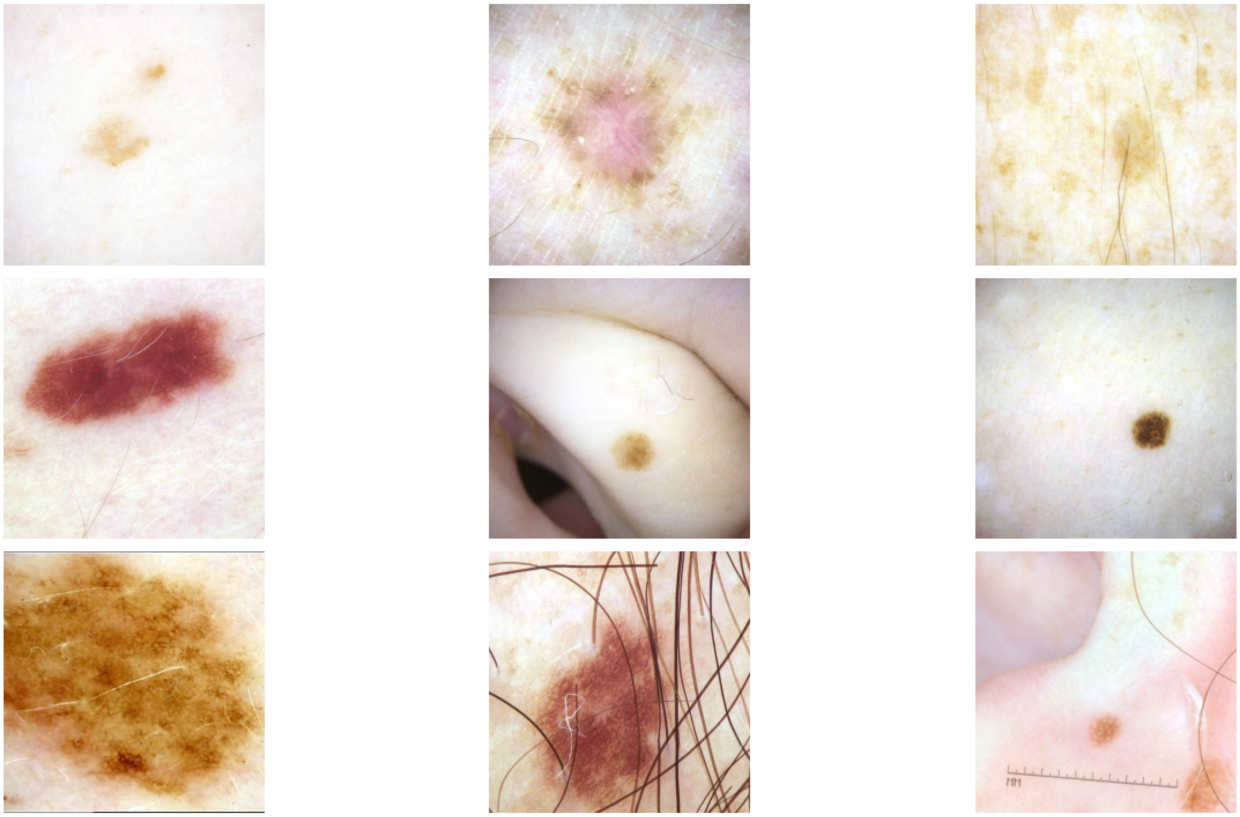
Here we see that comparatively malignant patients are of higher age and their average age is almost **57 years** whereas average age of the benign patients is close to **50**.



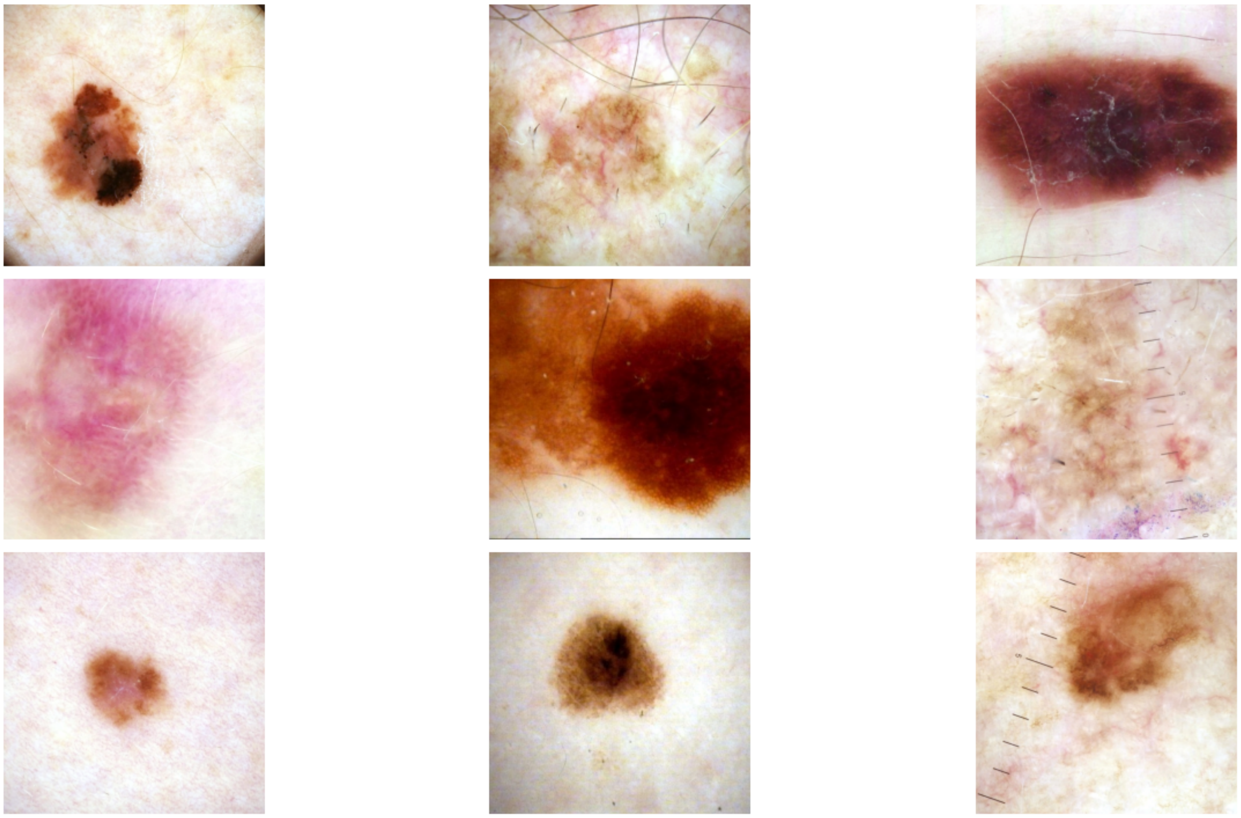
We can see that oral/genital and palms/soles are least likely to develop Melanoma disease whereas most of the time torso develops melanoma

Visualizing Some Images

Benign:



Malignant:



Data Processing

1. Balance the dataset to have 50 % malignant and 50 % benign images

2. Check for Missing Values.

3. Remove missing values.

4. Convert the categorical to numerical.

Training_samples - 966

Validation_sample - 230

Test Samples = 184

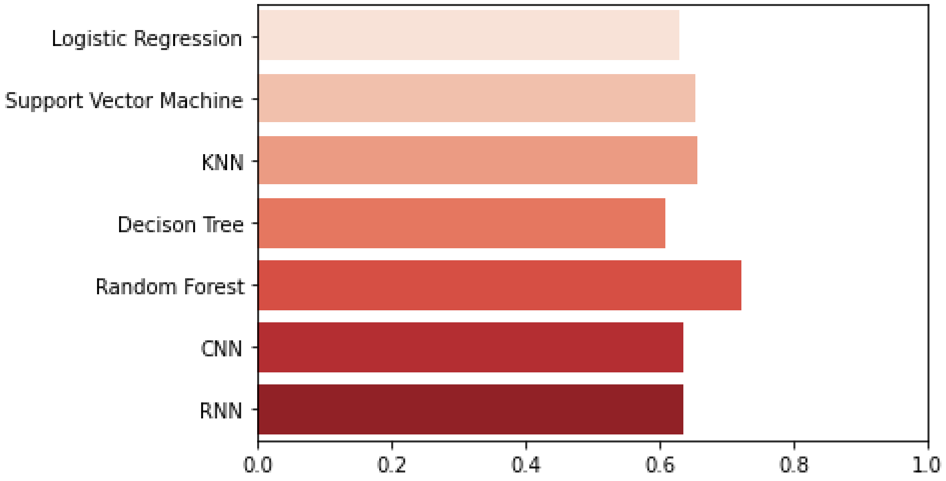
There are Three Different Evaluations Done.

1. Raw and augmented images.
2. Features from images extracted using Vg166 pretrained model on imagenet dataset
3. Tabular Dataset

Evaluation of Models

Images:

These results are trained on images of size **(512,512,3)** resized to **(224,224,3)** and then augmented and preprocessed to smaller size.



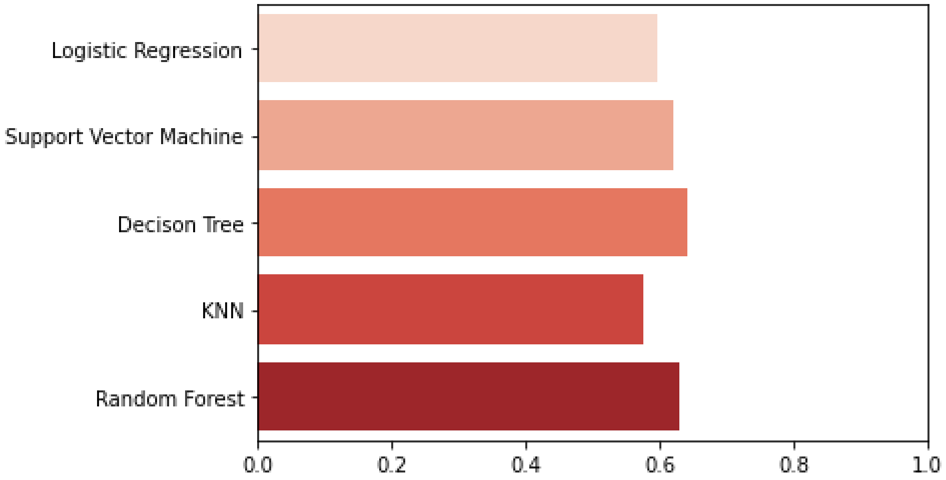
Most have test_accuracy of about 0,62 percent.

Tabular Data:

These results are trained on Attributes in csv file

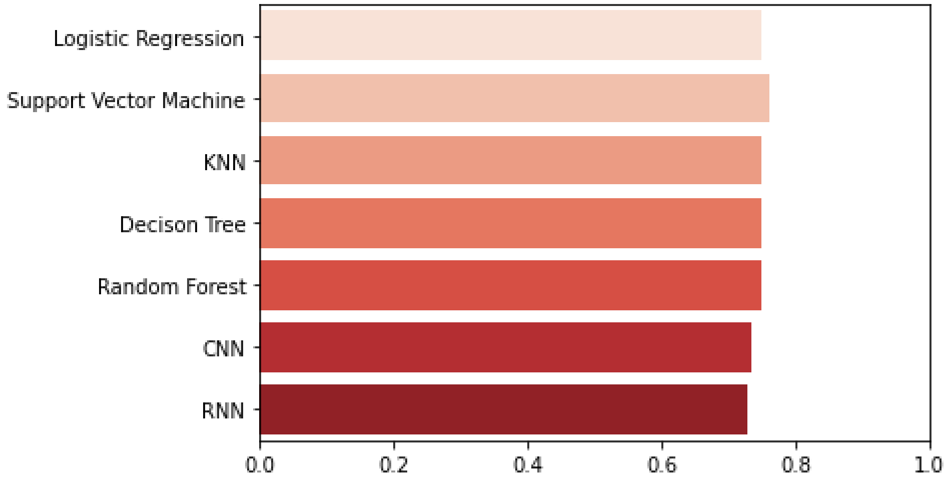
- sex
- anatom_head/neck
- anatom_lower extremity
- anatom_oral/genital
- anatom_palms/soles
- anatom_torso
- anatom_upper extremity
- Age

Encodings with anatom prefix are hot encodings of where the tumor is present. The most common location is torso as seen above in EDA.



Images Features Extracted From VGG16:

These results are trained on images of size **(512,512,3)** resized to **(224,224,3)** and then augmented and preprocessed through a pretrained model to get the features of the last layer and which is then used to predict the class.



Most have test_accuracy of about 0,75 percent.

Feature Extraction:

Instead of using object detection or segmentation we are using the features extracted at the last layer of pretrained vgg16 on imagenet dataset. It serves the same purpose and makes it easy for us to identify the Image.

Cost Function:

The cost function used in RNN and CNN is Binary cross-entropy which is a suitable loss function for our problem for these reasons:

1. **Binary problem:** Since your goal is to predict whether an image is benign or malignant, you have a binary classification task. Binary cross-entropy is specifically

designed for problems with two classes, making it an appropriate choice for your problem.

2. **Probabilistic interpretation:** Binary cross-entropy loss measures the difference between the predicted probabilities and the true binary labels, which is suitable for this problem because the output of a CNN for binary classification is typically a sigmoid activation function that returns a probability between 0 and 1. This probability represents the likelihood that an image is malignant.
3. **Penalizes incorrect predictions:** Binary cross-entropy loss heavily penalizes predictions that are far from the true label. This property encourages the model to output probabilities close to the true label, improving classification performance.
4. **Gradient-based optimization:** CNNs are typically optimized using gradient-based techniques like stochastic gradient descent (SGD) or its variants. Binary cross-entropy is a smooth, differentiable function that provides informative gradients to update the model's parameters, facilitating the optimization process.
5. **Robustness**: Binary cross-entropy is less sensitive to imbalanced datasets, which can occur in medical imaging tasks like predicting malignancy. This robustness is helpful because it reduces the chance of the model being biased towards the majority class.

Binary cross-entropy is a good choice for your problem because it's designed for binary classification tasks, provides a probabilistic interpretation, penalizes incorrect predictions, works well with gradient-based optimization, and is robust to class imbalance. This makes it a suitable loss function for training a CNN to predict whether an image is benign or malignant.

KMeans on Unlabelled Data:

I used KMeans Clustering on unlabelled images. The input is a raw image of size 224,224,3.

Number of Clusters was set to 2.

Metric of Determination: Silhouette

Silhouette score: 0.5170565367581895

A score of 0.517 means that the clustering has produced reasonably well-separated clusters, but there is still room for improvement.

Transfer Learning using Resnet50:

To improve performance, I used resnet50 with weights from imagenet dataset and fine tuned it on my dataset. The inputs are raw images of size 224,224,3 which are processed and augmented.

Test Accuracy: 0.77.

This is the best model. On raw images it performed better and gave an accuracy of 0.77 compared to other models with an average of 0.65 on raw images.

Ensembling Models

Kaggle's Melanoma Classification competition provides both image data and tabular data about each sample. Our task is to use both types of data to predict the probability that a sample is malignant. How can we build a model that uses both images and tabular data?

Three ideas come to mind.

1. Build a CNN image model and find a way to input the tabular data into the CNN image model

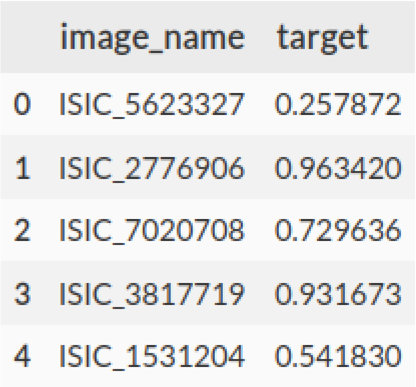
2. Build a Tabular data model and find a way to extract image embeddings and input into the Tabular data model

3. Build 2 separate models and ensemble

We will do the third one.

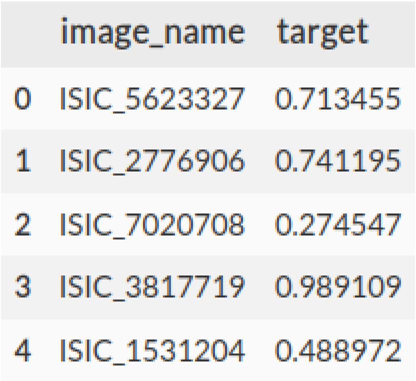
We are ensembling 2 model results based on weighted average

Results from XGBoost (Decision Tree):



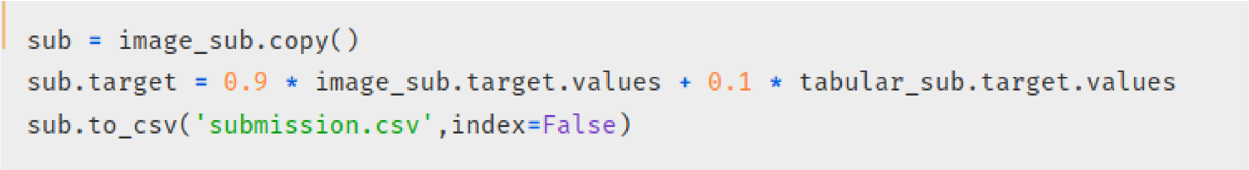
	image_name	target
0	ISIC_5623327	0.257872
1	ISIC_2776906	0.963420
2	ISIC_7020708	0.729636
3	ISIC_3817719	0.931673
4	ISIC_1531204	0.541830

Results from Transfer Learning using Resnet50:



	image_name	target
0	ISIC_5623327	0.713455
1	ISIC_2776906	0.741195
2	ISIC_7020708	0.274547
3	ISIC_3817719	0.989109
4	ISIC_1531204	0.488972

Combining Both based on Weighted Average Resnet Results are given more weight. This gives us better predictions by ensembling these 2 models



```
sub = image_sub.copy()
sub.target = 0.9 * image_sub.target.values + 0.1 * tabular_sub.target.values
sub.to_csv('submission.csv',index=False)
```