# Written Report
Machine Learning Project - CSC8635

Muzaffer Senkal - 210351491

21/01/2022

## 1   Introduction

Any technologies that can give more efficient, or faster analysis are very valuable. In recent decades, scientist and software developers are working together to improve healthcare technologies. Artificial Intelligence is playing huge role in these studies. Many studies have already surpassed human achievement, and many doctors use these systems as a decision mechanism, or diagnose the disease. Especially in cancer diagnosis or early diagnosis, artificial intelligence models provide great benefits to doctors in terms of speed and accuracy. Therefore, in this study, a model will be developed that can benefit the health field by using machine learning methods for skin cancer, which has been quite common in recent years.

## 2   Business Understanding

Early diagnosis plays a critical role in disease treatment processes in the healthcare. One of these diseases is skin cancer. In case of early detection, it contributes greatly to the recovery process of the individual. This detection is made with the Biopsy method [1] , which is the conventional method today. A sample must be taken, and the patient's skin is scraped off while taking this sample. Getting lab results is time consuming and also it is painful to take samples from the patient. However, today's research shows that if a good quality image of the diseased area is taken, it is possible to detect it using machine learning and image processing methods.

### 2.1   Business Objectives

There are many types of skin cancers. These three types that are most common in society; basal-cell skin cancer (basal-cell carcinoma) (BCC), squamous-cell skin cancer (squamous-cell carcinoma) (SCC) and malignant melanoma [2]. Therefore, it is very difficult to identify them from the images. Thus, this project objective is to develop machine learning models that can classify correctly skin lessions. This project will be carried out using machine learning methods and will be evaluated using common model evaluation techniques.

### 2.2   Data Mining Goals

Before starting the project, defining data mining goals contributes significantly to project progress. Accuracy is usually evaluated in classification problems. It is the ratio of the number of correct predictions to the total number of input samples. Accuracy will be used in the training and evaluation. In addition, success values should be close to the success average in the literature [3]. Therefore, the following criteria needs to be met;

- Success will be evaluated according to the accuracy and must be 85% and above.
- The model will be served with a web interface.

## 2.3 Project Tools

The most important feature of scientific projects is reproducibility, so it is necessary to work on this in a systematic and organized way. Therefore, the Project Template structure which is implemented in R will be adapted to the Python language and applied in this project. In addition, the project will be carried out within the framework of the **CRISP-DM** methodology, which is widely used in the data science industry.

**Git** will allows us to tracks the changes for reproducibility. It is an open source version control system that will speed us up as we develop our projects, large or small, and help us maximize efficiency. The versioning can improve the reproducibility of our scientific analyses. This study was carried out using Github which is a web-based storage service for projects that use Git as a version control system.

The lession images are approximately 3 gigabytes. It will require a lot of computational power, and it will take a lot of time to train the deep learning model on the local computer with this data. Therefore, a GPU machine is required. **Google Colab** is a free Jupyter notebook environment that runs in the cloud. Google lets us use their dedicated GPUs and TPUs for machine learning projects, so we will accelerate our project with Google Colab.

# 3 Data Understanding

The **HAM10000** dataset ("Human Against Machine with 10000 training images")[4] released by Philipp Tschandl, Cliff Rosendahl, Harald Kittler includes totally 10015 dermatoscopic images from different populations. This data set is prepared for academic machine learning studies. More than 50% of lesions have been confirmed by pathology, while the ground truth for the rest of the cases was either follow-up, expert consensus, or confirmation by in-vivo confocal microscopy. The ground truths of more than half of the dataset was provided with pathology confirmation, the rest was provided by confirmation methods such as follow-up, expert consensus, or in-vivo confocal microscopy. And also there are csv files which includes comprehensive metadata and pixel values of images.

### 3.0.1 Data Scheme

- **lesion_id:** unique id of lession

Example:

`"AM_0000118"`

- **image_id:** image id of lession

Example:

`"ISIC_0027419"`

- **dx:** short for diagnosis (for the patient

Possible Values:

`"bkl","nv", "df", "mel", "vasc", "bcc", "akiec"`

- **dx_type:** how the diagnosis was made

Possible Values::

`"histo", "consensus" ,"confocal", "follow_up"`

- **age:** The age of patient

Possible Values:

`"80.0"`

- **sex:** The sex of patient

Example:

```
'male', 'female', 'unknown'
```

- **localization:** the part of body

Possible Values:

```
'scalp', 'ear', 'face', 'back', 'trunk', 'chest', 'upper extremity', 'abdomen', 'unknown', 'lower extre
```

## 3.1 Abbreviations Values

In the dataset, most categorical variables are abbreviated. The abbreviations are as follows;

### 3.1.1 Type of skin disease:

- **nv:** Melanocytic nevi
- **mel:** Melanoma
- **bkl:** Benign keratosis-like lesions
- **bcc:** Basal cell carcinoma
- **akiec:** Actinic keratoses
- **vasc:** Vascular lesions
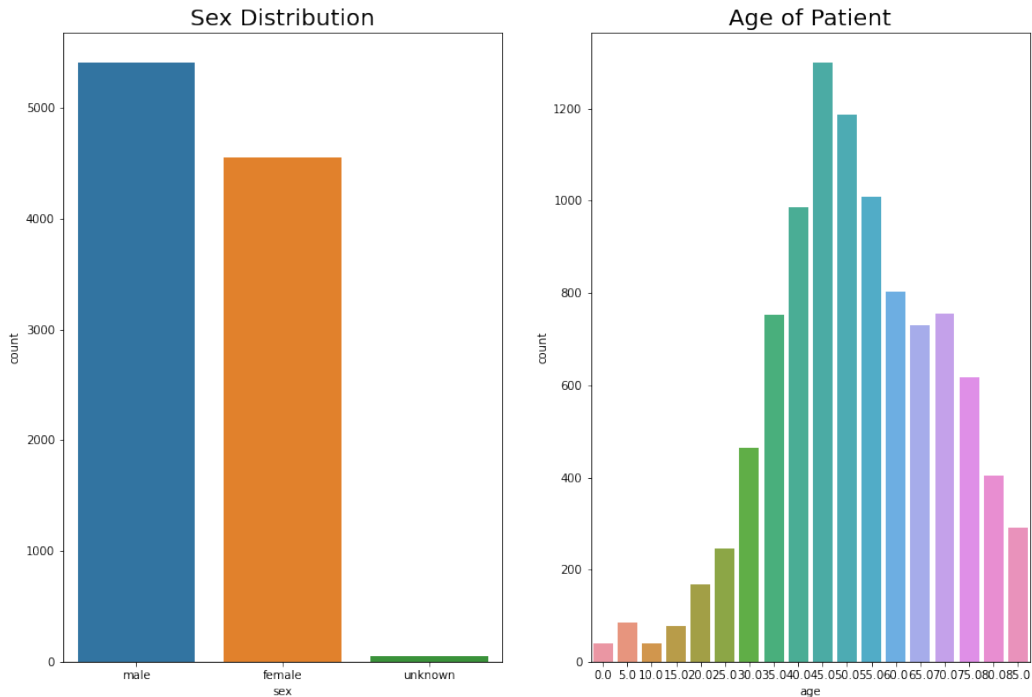- **df:** Dermatofibroma

### 3.1.2 Diagnosis methods

- **histo:** histopathology
- **follow_up:** follow up examination
- **consensus:** expert consensus
- **confocal:** confirmation by in-vivo confocal microscopy

## 3.2 Skin Images
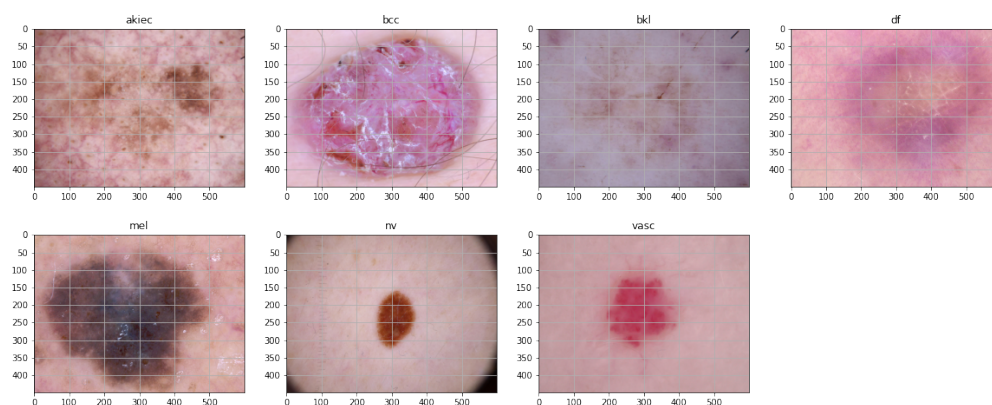
## 3.3 Exploratory Data Analysis

### 3.3.1 Univariate Analysis



- Skin diseases are more common in men than women.
- Skin diseases occur mostly in people aged 45 years and the probability of having skin disease is quite high in the middle age group. The number of samples increase dramatically after 25 years old. It remains stable between the ages of 60 and 70 and starts to decline again after the age of 75.
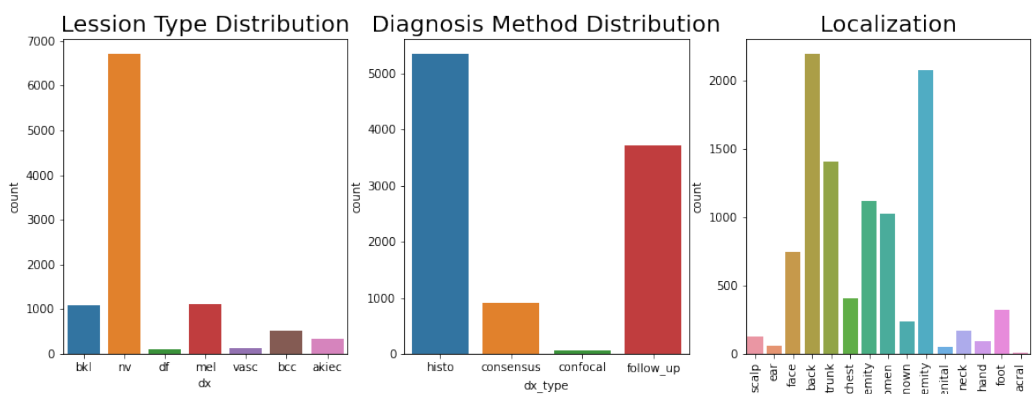


- Skin diseases occur mostly on the **"back"** and **"lower extremity"**(the part of the body from the hip to the toes) of the body and least on the **"acral"**.
- The most found disease among people is **"nv"** (Melanocytic nevi) while the least found is **"df"**(Dermatofibroma) .
- Diagnosis methods is mostly done by **"histo"** (Histopathologhy) and **"follow_up"** (Follow-up)

### 3.3.2 Bivariate Analysis



Localization vs Sex
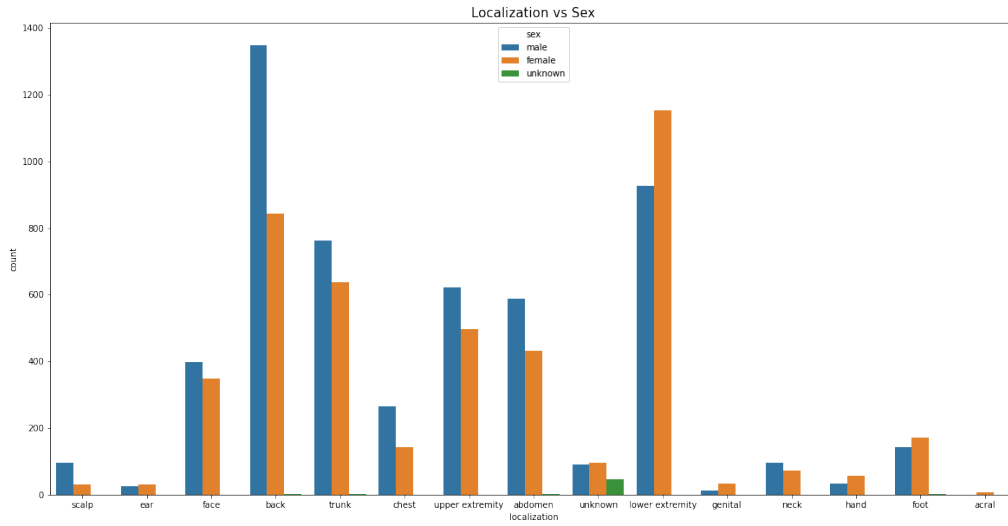
- In the univariate analysis, it was stated that males are generally more commonly afflicted with skin diseases. However, skin diseases seen in the parts of the body vary according to gender. As seen in the plot above, In the part of **skin extremity area**, **foot** and **hand**, women have more skin diseases than men in the .



Skin Disease type - Gender
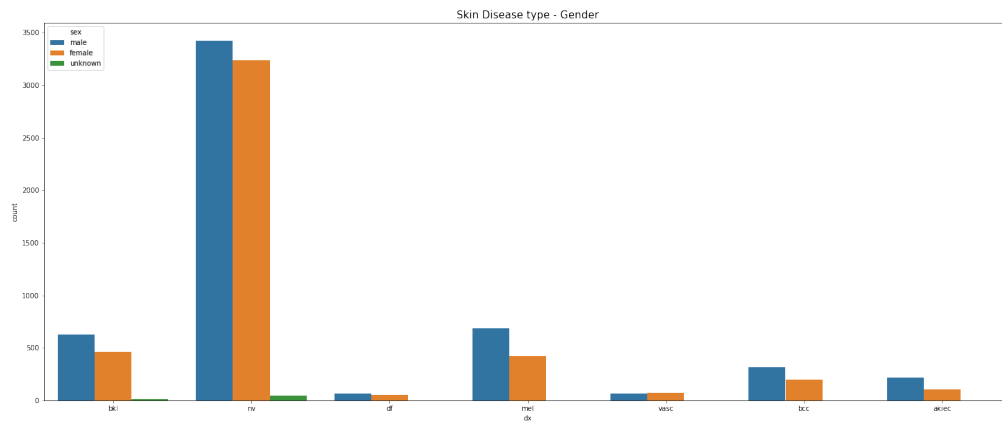
- Males and females are affected the most by Melanocytic nevi.

Skin Disease Type - Diagnosis method
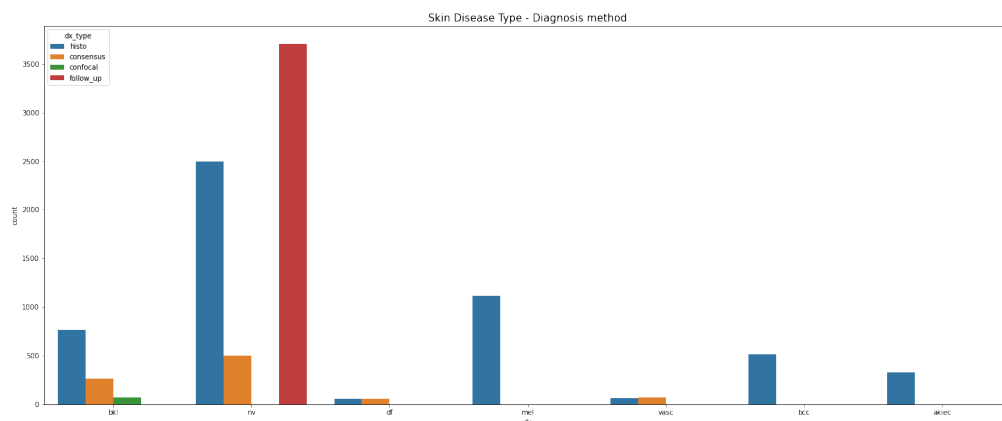
- It seems possible to determine all skin disease types with the **histopathology** method.
- When the **follow up examination** method was used, it was determined that the skin disease belonged only to Melanocytic nevi. In the same way **expert consensus** method was used to determine skin disease of the Benign keratosis-like lesions.
- Confirmation by in-vivo confocal microscopy method was used to determine disease types except mel, bcc and akiec.


Localization - Skin Disease Type
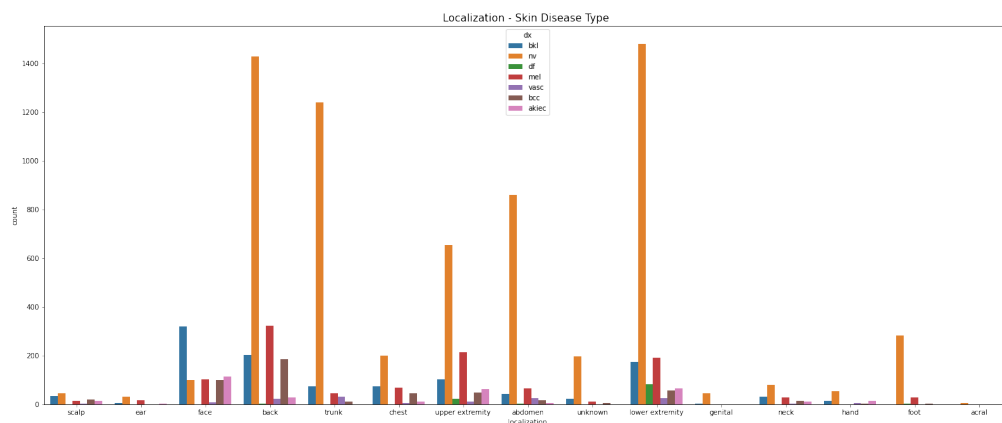
- The body part most affected by a benign keratosis-like lesion is the **face**.
- Body parts except face are infected the most by Melanocytic nevi.
- Only Melanocytic nevi and Melanoma skin diseases occur in the feet.
- Only Melanocytic nevi skin disease occurs in the genital area.

## 3.4 Data Quality

### 3.4.1 Missing Values

Looking at the dataset, only missing values are seen in the age column. However, we do not need to fill in these missing values for the deep learning model we will create.

# 4    Data Preparation

This stage includes all the operations that need to be done in order to move on to the model stage of the data set. Currently, our dataset is not ready for model training. For this, it is necessary to perform some data manipulation operations on the data.

**4.0.0.1    Lesion Name Abbreviations**    In the metadata dataset, lesion names were abbreviated. Abbreviations will be replaced with their original names so that they can be expressed more easily in the analysis.

- 'nv': 'Melanocytic nevi'
- 'mel': 'Melanoma'
- 'bkl': 'Benign keratosis-like lesions'
- 'bcc': 'Basal cell carcinoma'
- 'akiec': 'Actinic keratoses'
- 'vasc': 'Vascular lesions'
- 'df': 'Dermatofibroma'

**4.0.0.2    Target Column**    In the business understanding section, the business goal was determined as skin lesion classification. The model to be created will determine the type of lesion based on the image shown. Therefore, the target variable is the lesion type (dx). The target variable should be represented as a categorical variable.

**4.0.0.3    Adding Image Path**    In order to make input the lesion pictures with their ground truth into model, we should add the image file name and path in the data frame.

**4.0.0.4    Train - Test Split**    To evaluate our model, the data should be separated into two parts; train and test. In the evaluation part, we will predict test data.

# 5    Modeling

It has been stated in the Business Understanding section, early detection of skin cancer is crucial. The objective is creating a machine learning model that can classify the lesion type given an image. In this section, models will be created using convolutional neural networks. Afterwards, the successes will be compared according to the metrics and the best model will be selected.

In order to create convolutional neural networks, we will use Keras[5], an open-source framework developed by Francois Chollet. It allows us to build complex models by writing a few lines of code. Furthermore, Keras includes best CNN architectures pre-trained on large datasets such as Resnet50, VGG19, Xception, so on.

## 5.1    Building Model

### 5.1.1  1. Custom CNN Architecture
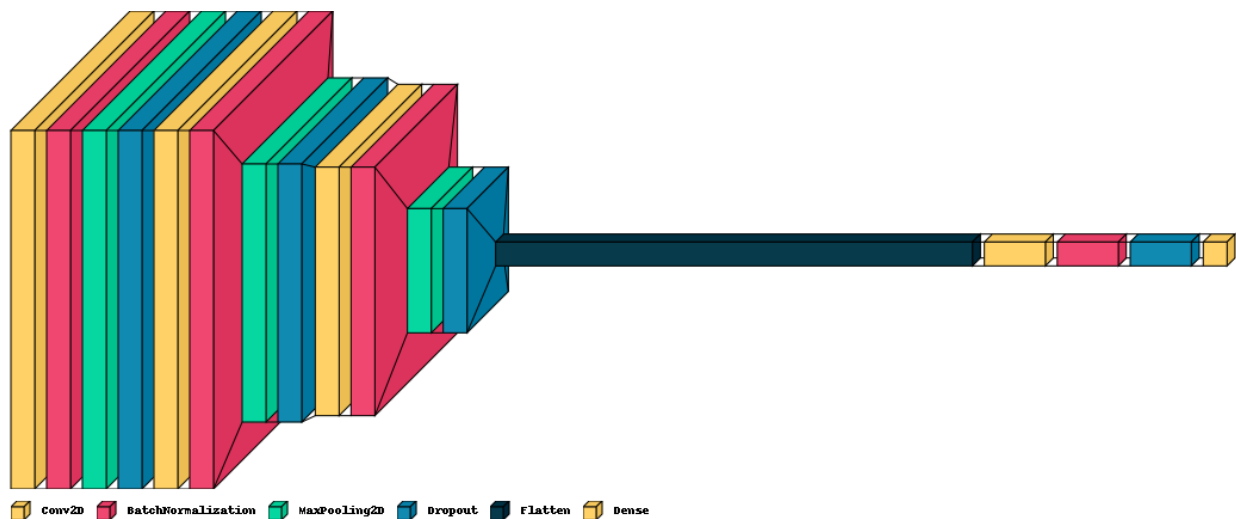


Firstly, Sequential layer is defined for our neural network which means a linear stack of layers between the input and output. Sequential layer in Keras allows us to build a model layer by layer using 'add()' function.

Our first layer is **Conv2D** layer which core block of the CNN. It will perform a dot product of input images, which are seen as 2-dimensional matrices. The activation function we will be using is Rectified Linear Activation. ReLu has been proven to work well in machine learning models. **BatchNormalization** is a layer that normalizes a mini-batch of data across all observations. **MaxPooling2D** is a pooling layer in CNN that performs dimensionality reduction that preserves locality in feature maps. After pooling layer we will use **Dropout** layer. In machine learning, regularization is a method to prevent overfitting. Thus, **Dropout** is a approach for regularization in neural networks.

We will add this quartet in our neural network 3 times in a row. In the last layer, we will add a **dense** layer which means the neurons of the layer are connected to every neuron of its preceding layer. Since we have 7 classes, the last layer has 7 neurons. Here **softmax activation** will be used as the activation function. Softmax function outputs a vector that represents the probability distributions of predict probability of classes.

### 5.1.2  Transfer Learning

The idea of **transfer learning** was shaped by the learning paradigm of human. People have an inherent ability to transfer knowledge across tasks. We use the knowledge we learn in one job in another job, so transfer learning in deep learning was inspired by this motivation. Transfer learning is the reuse of a model that has previously learned for another problem on a new problem. A model uses the knowledge learned from a previous problem to increase prediction about a new problem.

Three of the most popular architectures pre-trained on the ImageNet dataset were selected. ImageNet dataset includes over 14 million images sorted in 1000 categories [6].

- **Resnet50:** ResNet stands for Residual Network, and it has many variants that run on the same concept but have different numbers of layers. It is introduced by Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun [7]. As the name suggests, ResNet-50 is a 50-layer deep convolutional neural network. It has 48 Convolution layers along with 1 MaxPool and 1 Average Pool layer.

- **Inceptionv3:** The Inception V3 deep learning model is a variant of the Convolutional Neural Networks framework. It was developed by Google's team at the company.

- **Xception:** It was developed by Google, and involves Depthwise Separable Convolutions. DSC are kind of convolutions much more efficient in terms of computation time [8].

### 5.1.3 Transfer Learning Fine Tuning

In transfer learning part was to change only the layers of the classification stage, keeping the knowledge that the network obtained when extracting feature in the previous task, from which we are loading ImageNet weights. With fine-tuning we are not limited to retraining only the classifier stage (i.e. the fully connected layers), but what we will do is retrain also the feature extraction part. At this stage, the Resnet50, Xception, and Inceptionv3 models are fine-tuned.

## 5.2 Model Training

### 5.2.1 Train-Validation Split

To observe the success of the model during training the data should separate into training and validation. However, the data is not balanced, so we should do a stratified random split of the data. We will split data into **80** percent for training, **20** percent for validation. The random state ensures that the splits that you generate are reproducible.

### 5.2.2 Dataset Augmentation

Data augmentation is a method for oversampling the data. It generates new samples of images using various image transformation techniques such as image rotation, brightness or flipping etc. The benefit of using data augmentation is to prevent overfitting and enable the model to learn better.

| Parameter | Value | Description |
|---|---|---|
| rotation_range | 15 | rotates the image up to the given value |
| rescale | 1./255 | scale the image between 0 and 1 |
| shear_range | 0.1 | applies shearing transformations |
| zoom_range | 0.2 | zoom in or zoom out up to the given value |
| horizontal_flip | True | flips the image in horizontal direction |
| width_shift_range | 0.1 | shifts images in horizontal direction |
| height_shift_range | 0.1 | shifts images in vertical direction |

#### 5.2.2.1 Callbacks

- **Early Stopping:** To prevent overfitting we will use the early stop technique. If the validation loss value does not improve during a certain epoch, the model stops training. This value was selected as 10 epochs.
- **Learning Rate Reduction:** The learning rate will be reduced when a accuracy does not increase for 2 steps
- **Model Checkpoint:** There are too many images and probably the training time will take too much time. If something happens during training, we have to start all over, so we will save our model weights when validation accuracy increased.

#### 5.2.2.2 Optimizer
For optimizer, a widely known Adaptive Moment Estimation called **Adam** was used. Adam is an optimization algorithm to update network weights iterative based in training data. According the comparison of other algorithms, it requires less memory and is efficient [9].
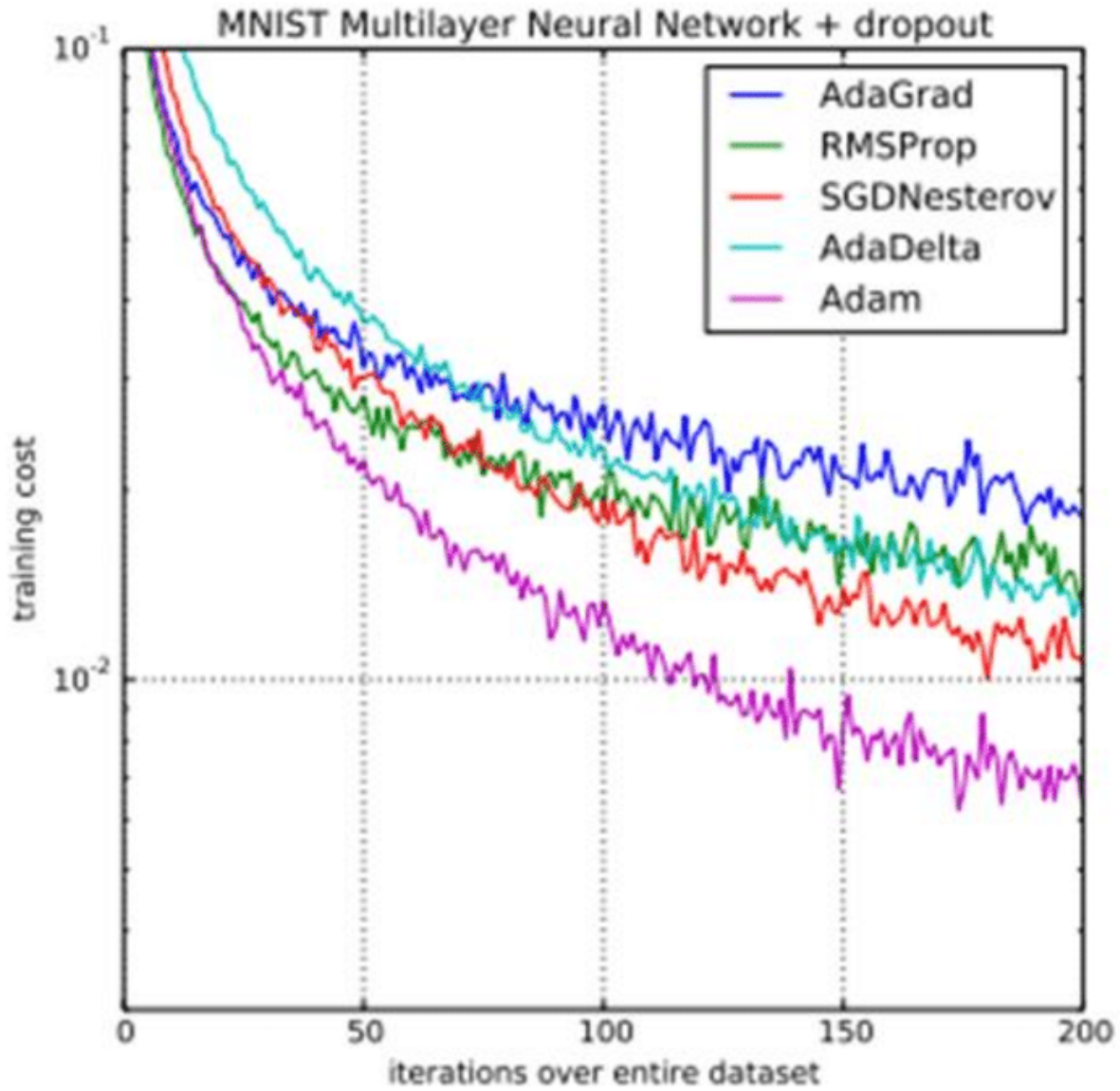
## 6   Evaluation

During the modeling phase, many models were developed for our business goal. At this stage, the success of these developed models will be compared and inferences will be made according to the criteria determined in our business target.

### 6.0.1 Model Training Evaluation
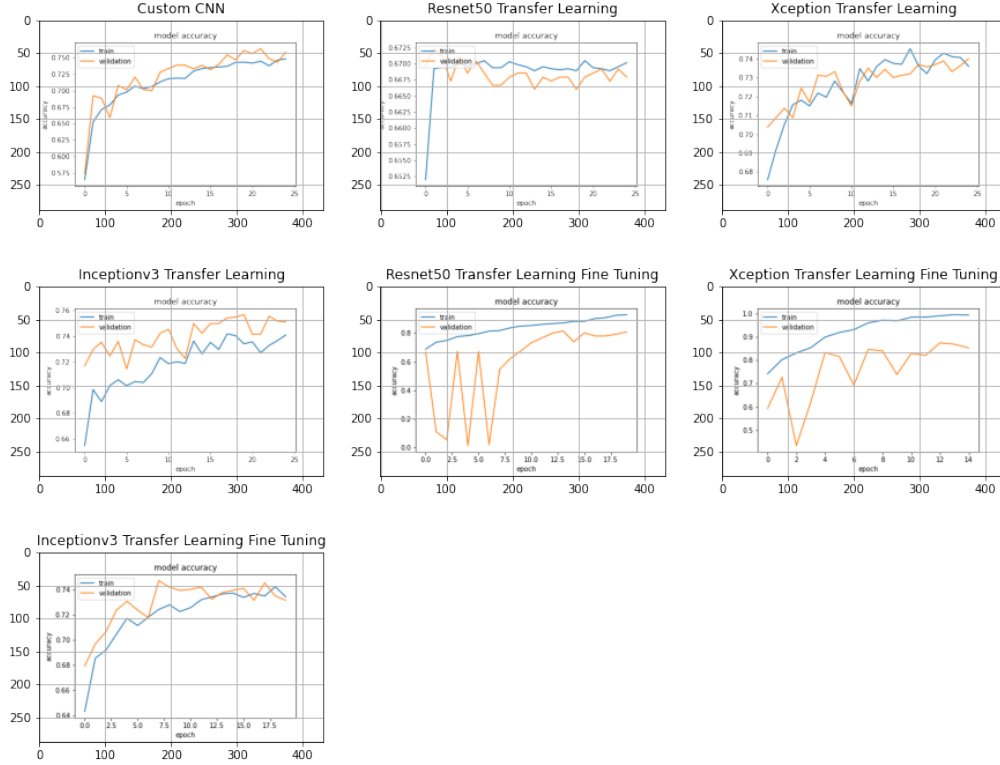


- The accuracy rate of our **custom cnn model** is approximately 75%. There is not much difference between the validation score and the training score, and there is no overfitting.
- The **Resnet50** architecture we use for transfer learning, could not close the success of our custom cnn architecture. The accuracy rate hovers around 67%. Deep fluctuations occurred in the first epochs of fine tuning. In the next epochs, the rates got closer and the training accuracy reached 93%.
- The success rate of the **Inceptionv3** architecture has increased to approximately 75%. There is an average of about 2% difference with the validation score. In Fine Tuning, this difference is greatly reduced.
- The accuracy of the **Xception** architecture is around 74%. There is a very small difference between the validation score and the training score. In fine tuning, the training score increased to 99% and the validation score to 85%. An overfitting has occurred in model training. However, when the validation score is considered, its success is quite high compared to other models.

**6.0.2   Test Data Evaluation**
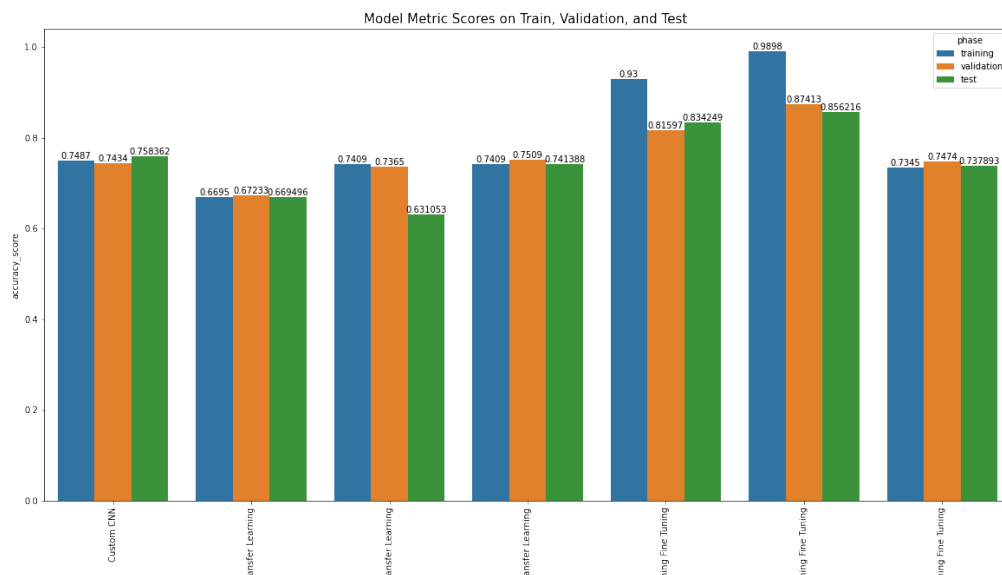


The graph above shows the success of the trained models on the training, validation,and test data. Accuracy metric value were calculated for each model. First of all, The success rate of the custom cnn model in the test set is slightly above the training and validation score. According to the scores, 2 models stand out; fine-tuned **Xception and Resnet50**. 2 models were overfitted during the training phase, but their success on validation and test data is above the success of other models. When the 2 models are compared to each other, the success of the **Xception** model on both validation and test set is higher than the Resnet50 model metrics.

**6.0.2.1   Confusion Matrix**   Predictions and groundtruths of all models on the test data are plotted through the confusion matrix below.
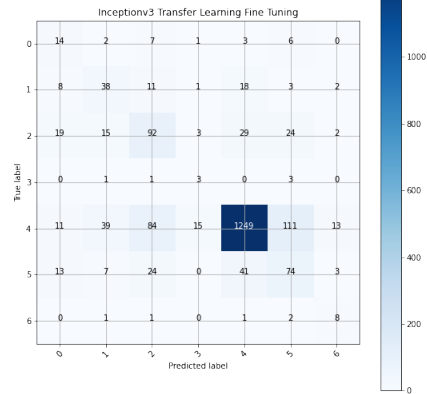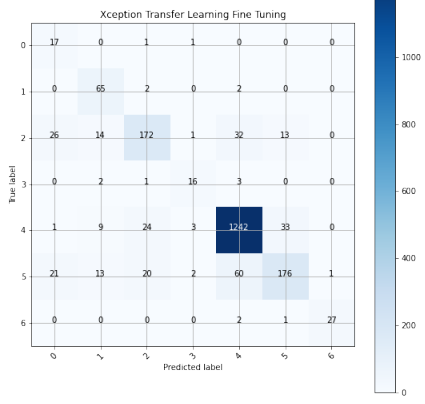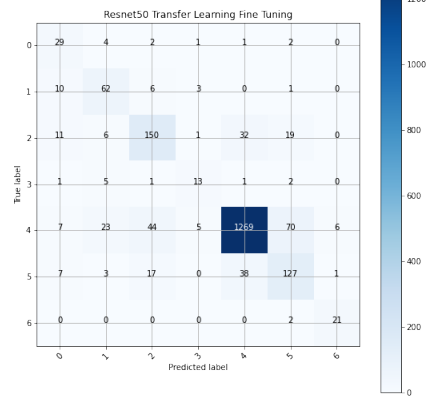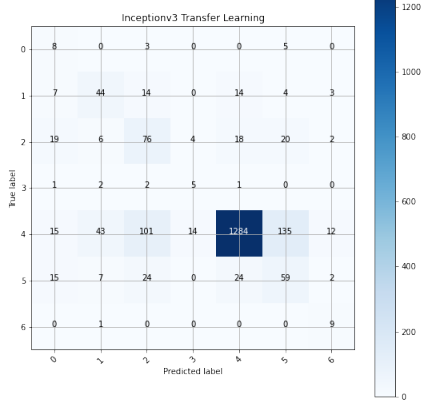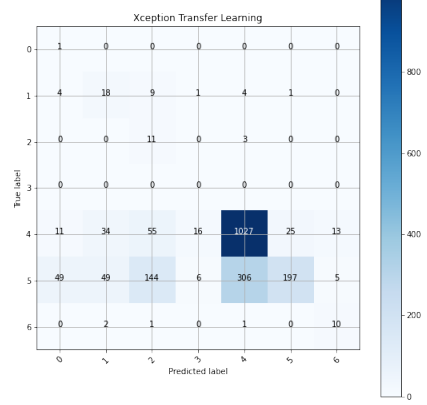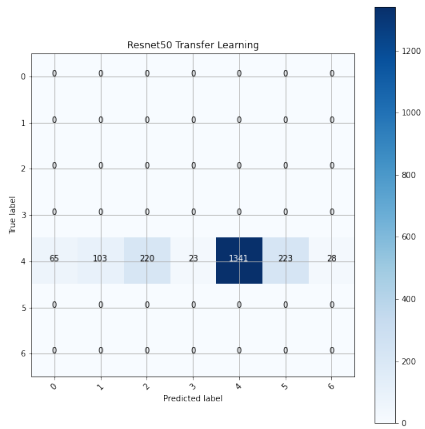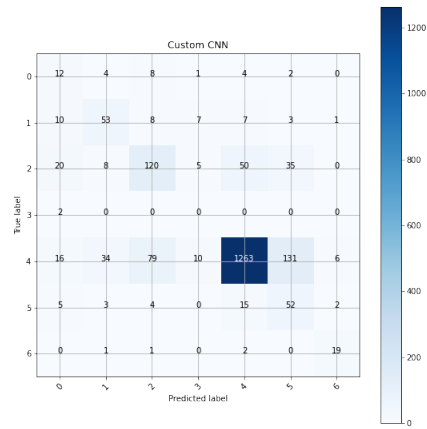
# 7 Deployment

Deployment is one of the most important stages. It is not possible for the study to reach its business goal without deployment. This section aims to put the whole work into practice. We will build an app that lets you upload images of skin lession and, with the help of a deep learning model, classify the image. Streamlit [10] was used to integrate the model into an interface. It is an open-source python framework for building web apps for Machine Learning.



# 8 Future Implications

The future implications on the work of this area are the creation of decision support mechanisms by integrating such studies into cloud systems. Having deep learning methods integrated into hospital systems such as PACS / HL7 is important for future diseases and epidemics.

# 9 Personal Reflection

The CRISP-DM data mining approach and Git were used to complete this project, which resulted in a high level of efficiency and repeatability. Also, versioning has greatly benefited deep learning projects. I tried to integrate the ProjectTemplate tool, which was used in previous data mining studies, into this project. But I noticed that it was lacking in model artifacts. For example, as in MLflow framework[11], model parameters, model weights and metadata can be kept in a structured structure.On the other hand, existing local computers are not sufficient for computer vision projects as deep learning models require large computational costs. For this, cloud compute services are needed to a great extent. A great deal of time was saved in model training using GPU machines from Google's Colab service. Finally, it is not correct to run deep learning models only once and compare their success. Many parameters should be changed and its success should be compared again and again.

## 9.1 References

[1] Jaleel, J. A., Salim, S., & Aswin, R. B. (2013). Computer Aided Detection of Skin Cancer. 2013 International Conference on Circuits, Power and Computing Technologies (ICCPCT). https://doi.org/10.110

9/iccpct.2013.6528879

[2] Wikipedia Contributors. (2021, December 12). Skin cancer. Wikipedia; Wikimedia Foundation. https://en.wikipedia.org/wiki/Skin_cancer#Classification

[3] Dildar, M., Akram, S., Irfan, M., Khan, H. U., Ramzan, M., Mahmood, A. R., Alsaiari, S. A., Saeed, A. H. M., Alraddadi, M. O., & Mahnashi, M. H. (2021). Skin Cancer Detection: A Review Using Deep Learning Techniques. International Journal of Environmental Research and Public Health, 18(10), 5479. https://doi.org/10.3390/ijerph18105479

[4] Tschandl, P., Rosendahl, C., & Kittler, H. (2018). The HAM10000 dataset, a large collection of multi-source dermatoscopic images of common pigmented skin lesions. Scientific Data, 5(1). https://doi.org/10.1038/sdata.2018.161

[5] Team, K. (2022). Keras: the Python deep learning API. Keras.io. https://keras.io/

[6] ImageNet. (2021). Image-Net.org. https://www.image-net.org/

[7] He, K., Zhang, X., Ren, S., & Sun, J. (2015). Deep Residual Learning for Image Recognition. ArXiv.org. https://arxiv.org/abs/1512.03385

[8] Chollet, F. (2016). Xception: Deep Learning with Depthwise Separable Convolutions. ArXiv.org. https://arxiv.org/abs/1610.02357

[9] Kingma, D., & Lei Ba, J. (2017). ADAM: A METHOD FOR STOCHASTIC OPTIMIZATION. https://arxiv.org/pdf/1412.6980.pdf

[10] Streamlit The fastest way to build and share data apps. (2013). Streamlit.io. https://streamlit.io/

[11] MLflow - A platform for the machine learning lifecycle. (2021). MLflow. https://www.mlflow.org/