People's Democratic Republic of Algeria
Ministry of Higher Education and Scientific Research

University M'Hamed BOUGARA – Boumerdes

Institute of Electrical and Electronic Engineering

Department of Electronics

Final Year Project Report Presented in Partial Fulfilment of the Requirements for the Degree of

MASTER

In **Electronics**

Option: **Computer Engineering**

Title:

Melanoma identification using convolutional neural networks

Presented by:

- **LOUIFI Akram**
- **SOUALMI Ameur**

Supervisor:

DR. CHERIFI Dalila

Registration Number:……/2018

Dedication

We dedicate this humble work

To

My parents who made me the man I am today.

To

My brothers and Sisters who never hesitated to help

To

My dear friends, both old and new ones

To

The most special Teacher I had the honor to be a student of

"Dr. Cherifi Dalila"

Akram

Dedication

To my beloved Mother, Father, and siblings

To the greatest teacher, my source of inspiration, Dr Cherifi

To my friends and the squad: Ramzi, Yacine, Saleh, yanis, said, and lokman f lghaybiya

To IGEE *students of 2013 promotion*

To ECN *students of 2010 promotion*

To everyone who helped me during my good and bad times

Finally to Akram, It was an honor to work with you.

Ameur

Acknowledgement

First, we thank Allah the Almighty for enabling us to realize this modest Work.

A special thanks to our supervisor Dr. Cherifi Dalila for her continuous help and immense support.

We also thank to Dr. Rachid Benlamri ,Dr. Emad Mohammed ,and Maria for their help and guidance.

We also thank the staff of I G E E for providing a convenient environment without forgetting all the ones that contributed in this document.

Abstract

Melanoma is an extremely dangerous type of skin cancer causing fatal incidences, it's also an increasing form of cancer around the world. Since the odds of recovering for the early-diagnosed cases is very high, early detection of melanoma is vital. Computer assisted diagnosis have been used alongside traditional techniques so as to improve the reliability of detecting melanoma.

In this project, a convolutional Neural network model designed from scratch as well as Transfer Learning using the pretrained model Inception *v3* are used in order to develop a reliable tool able to detect melanoma that can used by clinicians and individual users. The results using Inception *v3* model for dermoscopical images achieved the best results compared to our model. The results are compared to those of clinicians, which shows that the algorithms can be used reliably for the detection of melanoma.

Table of content

List of figures

List of tables

INTRODUCTION

Introduction

Malignant melanoma is the most dangerous type of skin cancer. 105.000 new cases of melanomas are estimated annually and 33.000 deaths (Parkin et al., 1999). Melanoma have a high mortality rates, mostly for the reason of the slim opportunities for early diagnosis and limited access to therapy (Siegel et al., 2015).The 5-year and 10-year relative survival rates for melanoma patients are 92% and 89% respectively (Miller et al., 2016). . This relatively high survival rates indicates why early detection of melanomas are highly important.

The rate of clinician's visual investigations gets at 65% of good detection at the very best. It is very difficult to distinguish some atypical lesions - which are benign - from melanoma because they have the same properties according to the well-known rules used by dermatologists.

In these last years, Computer assisted diagnosis have been a lively field of research. Many algorithms have been developed as a tool in order to help clinicians or even as a primary opinion for suspicious individual users. Many approaches were used (SVM, Very Deep Learning) to tackle this problem, Achieving moderate to satisfactory results.

The aim of this project is using Convolutional Neural Networks which work best for unstructured data .Also applying Transfer Learning which gained a lot of popularity for classification tasks because of the good performance as well as the reasonable training time.

CHAPTER 1

MELANOMA SKIN CANCER

1-Melanoma Skin Cancer

Melanoma, also known as malignant melanoma, is a type of cancer that develops from the pigment-containing cells known as melanocytes [1]. Other names for this cancer include malignant melanoma and cutaneous melanoma. Most melanoma cells still make melanin, so melanoma tumors are usually brown or black. But some melanomas do not make melanin and can appear pink, tan, or even white [2]

Melanoma is a disease in which malignant cells form in melanocytes. Melanocytes are the cells that produce melanin, which gives the colour to the skin. They make more pigment when the skin is exposed to sun or artificial light which causes the skin to darken [3]. The most common types of melanoma are superficial spreading melanoma, nodular melanoma, acral lentiginous melanoma, and lentigo maligna melanoma [4].

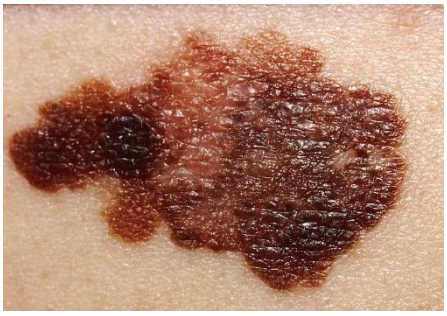


Figure 1 : Melanoma on a patient's skin, National Cancer Institute

Melanoma is the most dangerous type of skin cancer. Globally, in 2012, it newly occurred in 232,000 people [5]. In 2015 there were 3.1 million with active disease which resulted in 59,800 deaths [6]. Australia and New Zealand have the highest rates of melanoma in the world [5]. There are also high rates in Northern Europe and North America, while it is less common in Asia, Africa, and Latin America. In Table 1, the number of incidences of melanoma, mortality and 5-year prevalence for some countries of Europe and Australia are presented.

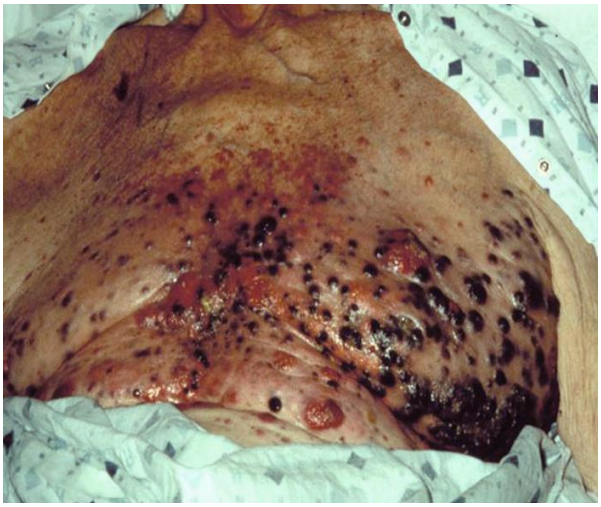


Figure 2: Extensive malignant melanoma on a person's chest .

1.1 Causes:

The primary cause of melanoma is ultraviolet light (UV) exposure in those with low levels of skin pigment, the UV light may be from either the sun or from other sources, such as tanning devices [5].

1.1.1 Ultraviolet light

Studies suggest that exposure to ultraviolet radiation is one of the major contributors to the development of melanoma. Occasional extreme sun exposure (resulting in "sunburn") is causally related to melanoma [7]. Ultraviolet UVB light (wavelengths between 315–280 nm) from the sun is absorbed by skin cell DNA and results in a type of direct DNA dammage called cyclobutane pyrimidine dimers (CPDs). Thymine-thymine, cytosine-cytosine or cytosine-thymine dimers are formed by the joining of two adjacent pyrimidine bases within a DNA strand. Somewhat similarly to UVB, UVA light (wavelengths between 400–315 nm) from the sun or from tanning beds can also be directly absorbed by skin DNA (at about 100 to 1000 fold lower efficiency than UVB is absorbed) [8]

1.1.2 Genetics

A family history of melanoma greatly increases a person's risk because mutations in several genes have been found in melanoma-prone families. People with a history of one melanoma are at increased risk of developing a second primary tumor. A number of rare mutations, which often run in families, greatly increase melanoma susceptibility. Several genes increase risks. Some rare genes have a relatively high risk of causing melanoma; some more common genes, such as a gene called MC1R that causes red hair, have a relatively lower elevated risk [9].

Table 1: *Melanoma incidence, mortality and 5-year prevalence in Europe (Ferlay et al., 2012 & Bray et al.,2012)and Australia (Australian Institute of Health and Welfare, 2015 & Australian Bureau of Statistics, 2015)*

Country	Incidence		Mortality		5 – year prevalence
	Number	Rate per 100000	Number	Rate per 100000	
EU (27 countries) (Ferlay et al., 2012 & Bray et al.,2012)	82075	13,0	22000	2,2	323467
Denmark	1596	24,1	228	3,0	6443
Germany	16884	14,8	2671	2,0	66997
UK	14445	19,0	2195	2,6	57163
Norway	1506	25,3	325	3,1	6247
Sweden	2911	23,9	565	4,0	12088
Switzerland	2484	25,8	384	3,5	10357
Netherlands	4804	24,4	853	3,9	20223
Australia (Australian Institute of Health and Welfare, 2015 & Australian Bureau of Statistics, 2015)	12960	49,0	1617	6,5	48364

1.2 Symptoms

Early signs of melanoma are changes to the shape or color of existing moles or, in the case of nodular melanoma, the appearance of a new lump anywhere on the skin or spots which are changing in color, size or shape and those looking different than other spots on the skin, known as the ugly duckling sign. At later stages, the mole may itch, ulcerate or bleed [10] these are the most important warning signs of melanoma. Early signs of melanoma are summarized by the mnemonic "ABCDE":

- Asymmetry
- Borders (irregular with edges and corners)
- Color (variegated)
- Diameter (greater than 6 mm (0.24 in), about the size of a pencil eraser)
- Evolving over time

These classifications do not, however, apply to the most dangerous form of melanoma, nodular melanoma, which has its own classifications:

- Elevated above the skin surface
- Firm to the touch
- Growing

Metastatic melanoma may cause nonspecific paraneoplastic symptoms, including loss of appetite, nausea, vomiting and fatigue [11].

1.3 Staging

Melanoma stages (5 year survival rates) [12]:

Stage 0: Melanoma in situ (Clark Level I), 99.9% survival.

Stage I / II: Invasive melanoma, 89–95% survival.

- **T1a:** Less than 1.0 mm primary tumor thickness, without ulceration, and mitosis < 1/mm2.
- **T1b:** Less than 1.0 mm primary tumor thickness, with ulceration or mitoses ≥ 1/mm2.
- **T2a:** 1.01–2.0 mm primary tumor thickness, without ulceration.

Stage II: High risk melanoma, 45–79% survival

- **T2b:** 1.01–2.0 mm primary tumor thickness, with ulceration.
- **T3a:** 2.01–4.0 mm primary tumor thickness, without ulceration.
- **T3b:** 2.01–4.0 mm primary tumor thickness, with ulceration.
- **T4a:** Greater than 4.0 mm primary tumor thickness, without ulceration.
- **T4b:** Greater than 4.0 mm primary tumor thickness, with ulceration.

Stage III: Regional metastasis, 24–70% survival

- **N1:** Single positive lymph node
- **N2:** Two to three positive lymph nodes or regional skin/in-transit metastasis
- **N3:** Four positive lymph nodes or one lymph node and regional skin/in-transit metastases

Stage IV: Distant metastasis, 7–19% survival

- **M1a**: Distant skin metastasis, normal LDH
- **M1b:** Lung metastasis, normal LDH
- **M1c:** Other distant metastasis or any distant metastasis with elevated LDH

1.4 Diagnosis

Visual inspection is the most common diagnostic technique. Moles that are irregular in color or shape are typically treated as candidates. To detect melanomas (and increase survival rates), it is recommended to to regularly examine moles for changes (shape, size, color, itching or bleeding) and to consult a qualified physician when a candidate appears [13].

1.4.1 ABCD:

ABCD is an acronym for Asymmetry, Border, Color and Dermoscopic structure. It was developed with lay people in mind, without the equipment of dermatoscopes and it is only defined for lesions larger than 6 mm.The rule is described in Table 2. A suspicious lesion is motivated by a score of 4.74–5.45, and anything higher than 5.45 suggests melanoma [14].

Table 2: *ABCD Rule of dermoscopy (Soyer H.P. et al.,* 2007)

		points	Weight factor	Sub-score range
Asymetry	Complete symetry	0	1.3	0.-2.6
	Asymmetry in one Axis	1		
	Asymetry in two axes	2		
border	Eight segments, one point abrupt cut-off of pigment	0-8	0.1	0.08
color	One point for each color : White; Red; light brown; Dark brown; black ; blue gray	1-6	0.5	0.5-3
Dermoscopic structures	One point for every structure: pigment network; structureless areas; dots; globules; Branched streaks	1-5	0.5	0.5-2.5
Total score range:				1-8.9

1.4.2 7-Point Checklist:

The seven-point checklist includes three major criteria (atypical pigment network, atypical vascular pattern, blue-whitish veil), scoring 2 for each criterion, and four minor criteria

(irregular streaks, irregular pigmentation, irregular dots and globules, regression structures), with a point of 1 for each[15]. Total score equal or higher than 3 indicates melanoma. In Table 3, the characteristics with relevant scores are presented.

Table 3: *7-point checklist method (Soyer H.P. et al., 2007)*

Criteria	7-point score
Major	
Atypical pigment network	2
Blue-whitich veil	2
Atypical vascular pattern	2
Minor	
Irregular streaks	1
Irregular pigmentation	1
Irregular dots/globules	1
Regression structure	1

1.4.3 3-Point Checklist

In 2003, during the Virtual Consensus Net Meeting on Dermoscopy (CNMD) a simple algorithm, called three-point checklist, was established to distinguish malignant from benign pigmented skin lesions [16].

The algorithm relies on the following criteria

1. Asymmetry in structure and/or in color in one or two axis of the lesion. The contour shape of the lesion does not impact on the symmetry.
2. Atypical network, defined as pigmented network with thickened lines and irregular distribution.
3. Blue-white structures, namely any white and/or blue color visible in the lesion, including blue-white veil, scar-like depigmentation, and regression structures such as peppering.

When two or three features are present, the lesion can be identified as malignant.

1.4.4 Menzies Method:

The Menzies Method is a simplified dermoscopy method for diagnosing melanomas, The Menzies method is based on 11 features which are scored as present or absent. This reduces the intra- and interobserver errors. This algorithm has been shown to enable primary care physicians to increase their sensitivity for the diagnosis of melanoma by 38% compared with standard clinical visualization. It was developed based on a training set of 62 invasive melanomas and 159 clinically atypical pigmented non melanomas were scored for 72 surface microscopic features [17].

Table 4 : *Menzies method (Soyer H.P. et al., 2007)*

Negative features
Point and axial symmetry of pigmentation
Presence of a single color
Positive features
Blue-white veil
Multiple brown dots
Pseudopods
Radial streaming
Scar-like depigmentation
Peripheral black dots−globules
Multiple colors (five or six)
Multiple blue/gray dots
Broadened network

1.4.5 Pattern analysis :

Pattern analysis is the method of diagnosis which is preferred by specialist dermatologists [16]. Pattern analysis is based on the analysis of the features that are present in the lesion [18]. The original pattern analysis has received criticism for low reproducibility and ambiguity of terms, resulting in the revised pattern analysis presented by Kittler et al. [19]. Revised pattern analysis defines five basic elements from which all dermatoscopic patterns are derived. The five elements are:

Lines: structures with parallel edges, with their length much greater than their breadth

Dots: circumscribed, small, round, indivisible pigmented structures with no length or breadth.

Clods: solid, circumscribed, diversely formed pigmented or unpigmented structures larger than dots, with length and breadth.

Circles: lines or collection of pigmented dots arranged sensibly equidistant from a common focal point that constitutes the center.

Pseudopods: short pigmented lines with a bulbous end.

In experienced hands, a dermascope is a fine aid for evaluating pigmented skin lesions. Helping to set aside benign examples and to accurately diagnose melanoma, or issues such as seborrhoeic keratoses.

1.4.6 Dermoscopy and skin cancer:

An experienced skin cancer specialist can often identify cancerous symptoms by eye but a dermascope adds an extra edge. A clarity of colour and structure that would be impossible for the naked eye, or a normal microscope to see. Colour in itself can be a diagnostic aid, especially the distribution of pigments. Precise analysis of shape, symmetry, asymmetry, provides further valuable information. Computer software can retain critical images for

further study, or future comparison. Good diagnosis remains in the hands of an experienced consultant, they are the key but they value the additional accuracy brought by a dermascope [20].

1.5 Treatment and Therapy:

The treatment options depend on the stage of melanoma and overall health. During its early stages, melanoma can be successfully treated with surgery alone. Other types of cancer treatment are effective for more advanced stages of melanoma. The main treatment for melanoma is surgical resection (removal). Wide local excision, a minor surgery, can usually cure early-stage melanoma [21].

Patients with stage III melanoma (cancer that has spread to the lymph nodes) may need lymphadenectomy, or surgery to remove the involved lymph nodes. For advanced melanoma, surgery is frequently combined with immunotherapy or targeted therapy [22].

CHAPTER 2

The Theory of Artificial Neural Networks

2 The Theory of Artificial Neural Networks

2.1 Introduction:

Neural networks are a one of most popular type of machine learning model. Convolutional neural network (CNN) is a special case of a neural network, which is the focus of this thesis. In this chapter, we will discuss regular neural networks, before getting around to CNNs and Transfer learning.

2.2 History and origins:

Neural networks were originally called artificial neural networks, because they were developed to mimic the neural function of the human brain. Pioneering research includes the threshold logic unit by Warren McCulloch and Walter Pitts in 1943 and the perceptron by Frank Rosenblatt in 1957 [23]. Even though the inspiration from biology is apparent, it would be misleading to overemphasize the connection between artificial neurons and biological neurons or neuroscience. The human brain contains approximately100 billion neurons operating in parallel [24] .Artificial neurons are mathematical functions implemented on more-or-less serial computers. Research into neural networks is mostly guided by developments in engineering and mathematics rather than biology.

2.2.1 Biological neural networks

A biological neural network consists of billions of similar cells. They have very similar structure and have an enormous amount of connection between each other, which these cells use to continuously communicate. The scientific term for such cell is a neuron, this terminology also used in artificial counterpart. Each biological neuron behaves as a switch, it either forwards the signal further if conditions meet or stays idle. The tremendous information processing capabilities of biological neural networks are the result of coordinated signals send by a huge number of neurons and immense level of interactivity in between neurons. Neuron cell is the main processing unit in a biological neural network. Neuron cells act like a switch, when stimuli reach a certain condition neuron propagates signal further on all outgoing connections. Figure 2.1 presents different components of the neuron. The incoming signals can come from other neurons or cells. The incoming signals are transferred through physical connections which are connected to neuron via synapses. Synapses are points of connection, the connection to neuron occurs through dendrite or directly to soma. Synapses are distinguished into two categories, the electrical and chemical. The main functional distinction between electrical and chemical, in the first case signal, delivered directly to the soma while in the second case the signal is transformed to a chemical signal and then back to electrical and then delivered to soma. Chemical synapse

has physical shapes in form of a cleft and this break direct connection to the soma. The cleft also called synaptic cleft. In order for a signal to pass from presynaptic side of the cleft to postsynaptic side of the cleft, the electrical signal is converted to the chemical signal substance at the presynaptic side and converted back to the electrical signal at postsynaptic side. This mechanism is chemical and is able to modulate the signal transferred by excreting different type or quantity of pulse[25].

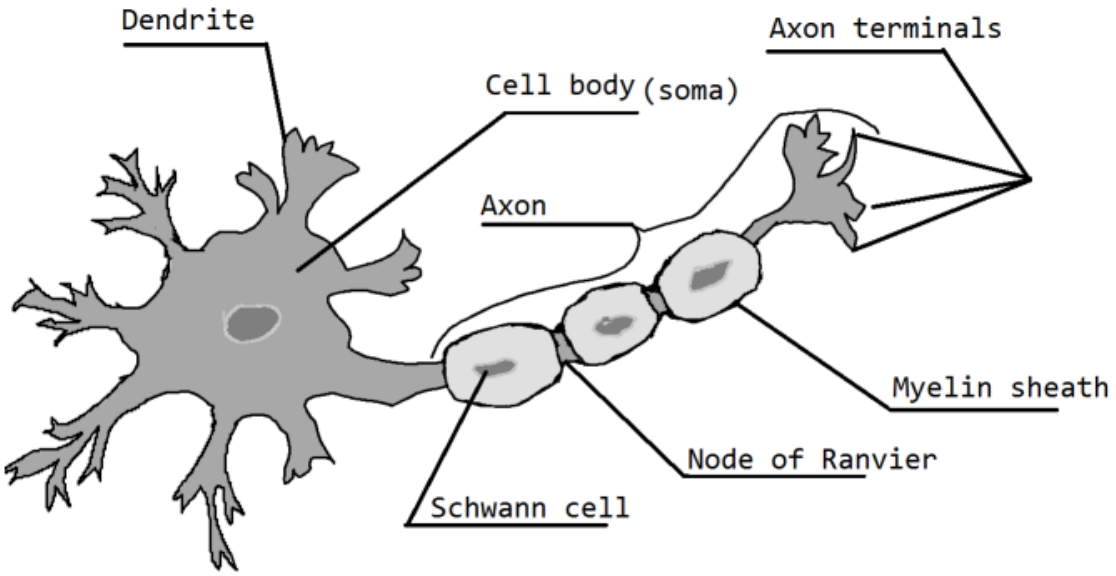


Figure 2.1: Neuron structure

2.2.2 From biological neural network to artificial neural network

Technical model is a strong simplification of the biological model. The abstraction of the biological neural network provided as follows: A neural network consists of a large number of entities called neurons, which act as nexus points of a large number of inputs. A function of a neuron is to act as a switch, which turns on when certain conditions met. The condition for activation is if the total of incoming stimuli at any time, except for refraction time, reaches a threshold the signal is propagated further. The signal sent in some cases allows amplitude or frequency modulation as a method of encoding information in the pulse. There are modified neurons, which have the function of receptors. Receptors are classified between primary with direct access to nervous system and secondary that are processed. In both classes of receptors, the information sent is encoded. Sensory neurons can form complex sensory organs, eyes, ears are such examples. There is a high level of interconnectivity between the neurons, in other words, a dense graph with neurons as nodes. Some connections are unidirectional and some

connections are bidirectional. Connections with chemical synapses provide weighting alike process for transmitted pulses.

The abstraction above reflects on elements in biological networks, which are transferred into the structure of an artificial network. The structure of an artificial network revolves around neuron construct, which has input and output a unique switch alike function [25].

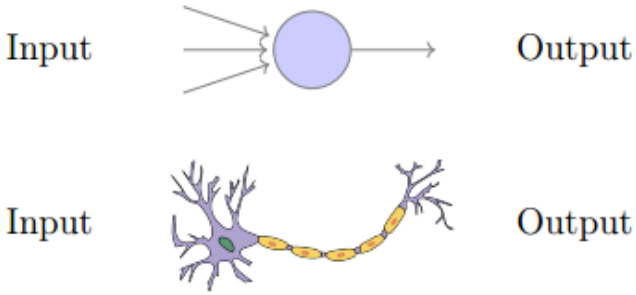


Figure 2.2: Comparison between neuron and its counterpart in Artificial Neural Networks.

2.3 Artificial Neural Networks:

2.3.1 The simple perceptron

The simple perceptron is the smallest possible ANN; it consists of only one neuron. The schematic representation can be seen in figure 2.3. In this figure, we see input variablesx_i, corresponding weights w_i the weighted sum of the inputs, an activation function ϕ, the output y and the bias b. Bias is an additional input to each neuron often represented as an input $x = 1$ and a weight w_0 ($w_0 = 0$). Unlike other applications bias is an often necessary thing in ANNs. The output y from the single perceptron is :

$$y = \varphi(\sum_{k=1}^{m} w_k x_k + \mathrm{b}) \qquad (2.1)$$

Including the bias in the summation, we get:

$$y = \varphi(\sum_{k=1}^{m} w_k x_k) \qquad (2.2)$$

Using a single perceptron the decision boundary is a hyperplane since we have linearity, (cf. linear SVM). With a single perceptron we may for example solve the AND and OR problem [26].

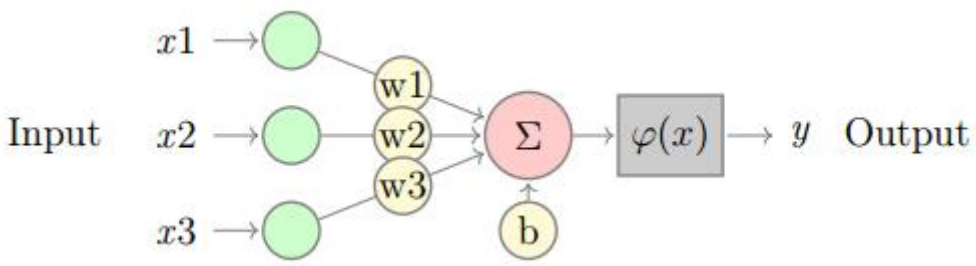


Figure 2.3: Schematic representation of a simple perceptron.

2.3.2 Multilayer Perceptron

Connecting many neurons we may construct a web, where one neurons output may be another neurons input. This may be done arbitrary, but this thesis will only work with so called feed-forward networks, ANNs that have no feed-back, i.e. input comes in from one side, propagates through the net and comes out as output at the other end. Such a network is called a Multilayer Perceptron (MLP), each step forward is called a new layer, and each layer is constructed of perceptrons. The term Deep Neural Networks or deep learning are also used, this originates from that adding layers increases the depth of the network.

Say we have a classification problem of three classes and a net looking like the one in figure 2.4. Then the largest output y_i is the most probable class for the input x.

$$y_i = \varphi_0(\sum_{i=0}^{5} w_{ij}\, \varphi_h(\sum_{k=0}^{4} \widetilde{w}_{jk}\, x_k)) \qquad (2.3)$$

as may be deduced from the graph. In classification it is common to use softmax on the output to get yi as probability for the input x to belong to class i. Softmax will be explained in section 2.4.4. When adding hidden layers the decision boundary is no longer limited to a hyperplane but can result in non-linear solutions. When adding hidden layers the decision boundary is no longer limited to a hyperplane but can result in non-linear solutions[26].

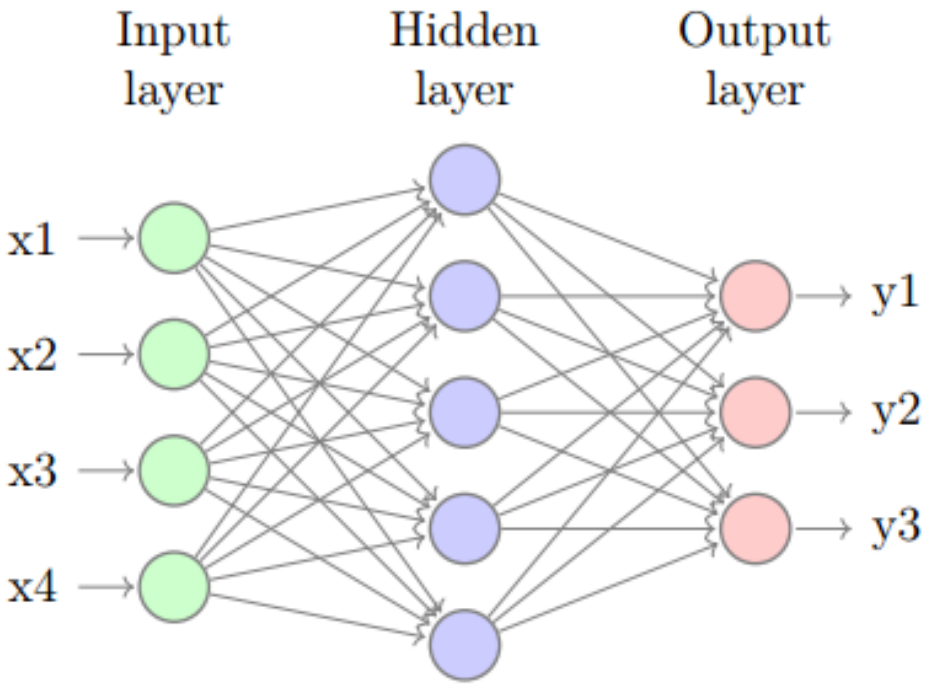


Figure 2.4: Schematic representation of a Multilayer Perceptron

2.4 Activations:

Output of each node is produced by the node's activation function φ that takes weighted inputs of the node as parameters transformed by a transfer function (Figure 2.5). The transfer function creates a linear combination of weighted inputs in order to feed them to the activation function. To approximate complicated functions, nonlinear activations are often used. The following sections briefly describe different nonlinear activation functions most commonly used in neural networks.

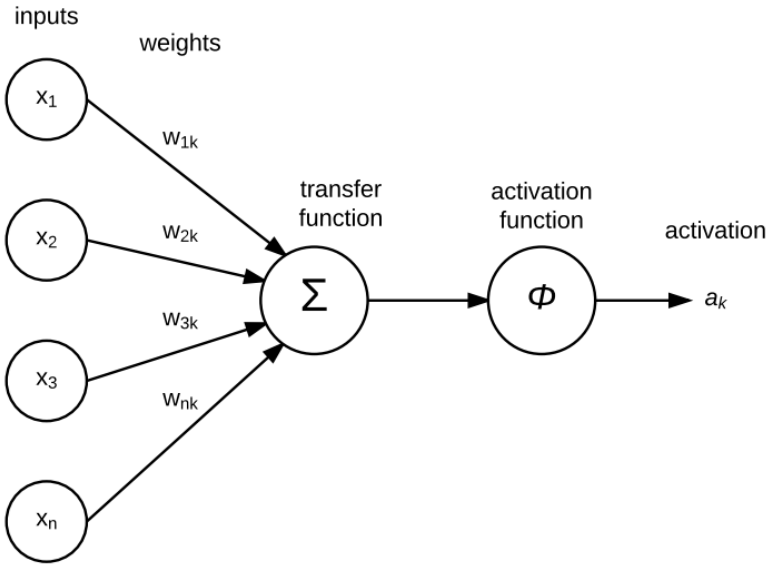


Figure 2.5: Placement of the activation function in the neural

Hyperbolic tangent:

One of the most popular activation functions is the hyperbolic tangent function (Eq 2.4). Input *x* is a weighted linear combination of the inputs of the node. This function works most effectively on inputs in range (0*;* 1), producing outputs in interval (-1*;* 1).

$$\tanh(x) = \frac{\sinh(x)}{\cosh(x)} = \frac{e^{x}-e^{-x}}{e^{x}+e^{-x}} = \frac{e^{2x}-1}{e^{2x}+1} \qquad (2.4)$$

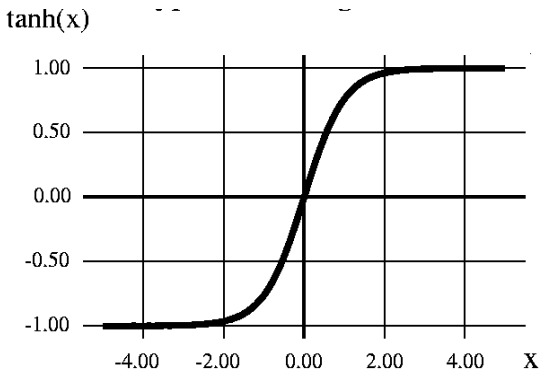


Figure 2.6 : Hyperbolic Tangent Function

Logistic Sigmoid:

The function (Eq 2.5) was named in 1844–1845 by Pierre François Verhulst, who studied it in relation to population growth. One of the reasons the sigmoid function is broadly used is the fact, the sigmoid function is differentiable at every point**.**

$$f(x) = \frac{1}{1+e^{-x}} \qquad (2.5)$$

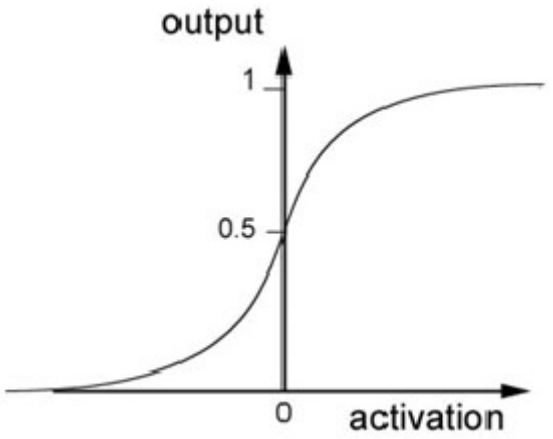


Figure 2.7 : Logistic Sigmoid Function

ReLU

Rectified linear unit's function (Eq 2.6) is used with the purpose to increase non-linearity of the network. Rectifying neurons are considered to be biologically more plausible than logistic sigmoid or hyperbolic tangent neurons [27]. They benefit from their simplicity, resulting in faster training and performance improvements in particular cases [28], and therefore often used in DNNs/CNNs. ReLU is given by the equation:

$$f(x) = \max(0, x) \qquad (2.6)$$

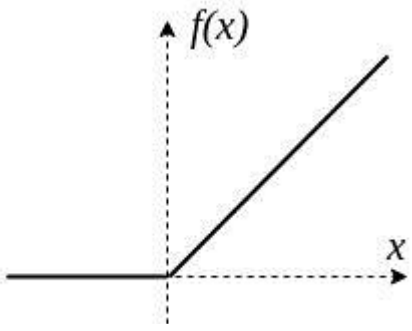


Figure 2.8 : ReLU Function

Softmax

The softmax activation function (Eq 2.7) is usually used in the last network layer, converting an arbitrary real value to posterior probability of the class c_k in range (0; 1):

$$p(c_k|x) = \frac{e^{a_k}}{\sum_{i=1}^{m} e^{a_i}} \qquad (2.7)$$

where *m* corresponds the number of output nodes (classes) and a_k is the activation value of *k*-th node:

$$a_i = \sum_{j=0}^{d} w_{ij}\, h_j(x) \qquad (2.8)$$

given *i*-th node's weights *wij* and the output of the previous layer *hj*(*x*).

2.5 Training a neural network

2.5.1 Supervised learning

Supervised learning is the machine learning task of getting function from supervised training data. The training data consist of a set of training examples. In supervised learning, each example is a pair consisting of an input object and a desired output value [29]. To make a network to give the right answers, it is necessary to train network, by changing weights (and neurons parameters). There are several methods for neuron network training. Basic learning algorithm of the neural network is a back propagation method, which uses the gradient descent algorithm.

2.5.2 Cost Function:

When modeling a certain function a common heuristic is the measure of how well this model approximates the function. This heuristic is often called cost- or loss function. Two common cost functions are mean squared error (MSE) and cross-entropy [30].

Mean Squared Error (MSE)

MSE is a measure of the quality of a predictor or estimator. It is the average squared distance between the predicted values and the true values, as described by (2.9).

$$\mathrm{MSE} = \frac{1}{N}\sum_{i}^{N}(\hat{y}_i - y_i)^2 \qquad (2.9)$$

In the case of neural networks the predicted values $\hat{y}_i$ are the outputs of the final layer and the true values y_i the output of the modeled function, N being the number of training samples.

Cross Entropy

The notion of entropy can also be useful as a cost function in classification problems, producing faster learning results than MSE [30]. The cross-entropy equation is also known as a negative log-likelihood [30]. Using the same notation as in MSE, the cross-entropy cost can be calculated by (Eq 2.10)

$$\text{Cross\_entropy} = -\frac{1}{\mathrm{N}}\sum_{\mathrm{i}}^{\mathrm{N}}\left(y \ln \hat{\mathrm{y}} - (1-\mathrm{y}) \ln(1-\widehat{\mathrm{y}})\right) \qquad (2.10)$$

2.5.3 Gradient Descent

Neural networks are often named gradient-based learning due to their use of gradients of the cost function for learning [31]. As is common practice in all areas of machine learning neural networks also try to reduce the cost function, but due to the non-linearity of neural networks most cost functions become non-convex and more fruitful results have been gained from moving in the negative direction of the gradient [31].

Given the cost function *C* as a function of the weights *w* and the biases *b* of the neural network,the gradient of *C* can be utilized in order to change the weights and biases to go fastest toward minimizing the cost function [30]. By introducing a scalar ŋ, called learning-rate, the weights and biases may be modified iteratively according to (eq 2.11.a/b)

$$\boldsymbol{b'} = \boldsymbol{b} - \eta \frac{\partial C(w,b)}{\partial b} \qquad (2.11.a)$$

$$\boldsymbol{w'} = \boldsymbol{w} - \eta \frac{\partial C(w,b)}{\partial w} \qquad (2.11.b)$$

Since the data volumes used for the training and testing of neural networks often are large it might not be possible to calculate an average change for the entire data set. Another method known as stochastic gradient descent may be applied. The stochastic method is based upon selecting a batch, or a subset, of the data and calculating the average change for that batch. This batch may be chosen to include any number of data points with the trade-off between noisy fluctuations and computation time and memory [30].

2.5.4 Backpropagation

A neural network is trained by selecting the weights of all neurons so that the network learns to approximate target outputs from known inputs. It is difficult to solve the neuron weights of a multi-layer network analytically. The back-propagation algorithm provides a simple and effective solution to solving the weights iteratively [32]. The classical version uses gradient descent as optimization method. Gradient descent can be quite time-consuming and is not guaranteed to find the global minimum of error, but with proper configuration works well enough in practice.

In the first phase of the algorithm, an input vector is propagated forward through the neural network. Before this, the weights of the network neurons have been initialized to some values, for example small random values. The received output of the network is compared to the desired output (which should be known for the training examples) using a loss function. The gradient of the loss function is then computed. This gradient is also called the error value. The error values are then propagated back through the network to calculate the error values of the hidden layer neurons. Finally, the neuron weights are updated by calculating the gradient of the weights and subtracting a proportion of the gradient from the weights. After the weights have been updated, the algorithm continues by executing the phases again with different input until the weights converge.

Another way of computing the updates is full batch learning, where we compute the weight updates for the complete dataset [32]. This is quite computationally heavy and has other drawbacks. A compromise version is mini-batch learning, where we use only some portion of the training set for each update [33].

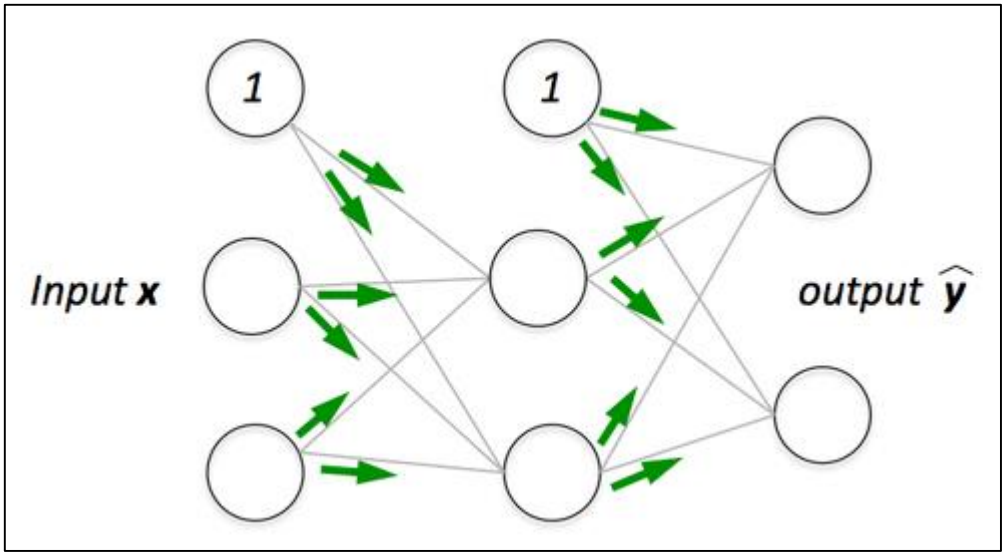


Figure 2.9: Forward propagation (passing the input signal through a network while multiplying it by the respective weights to compute an output) [34]

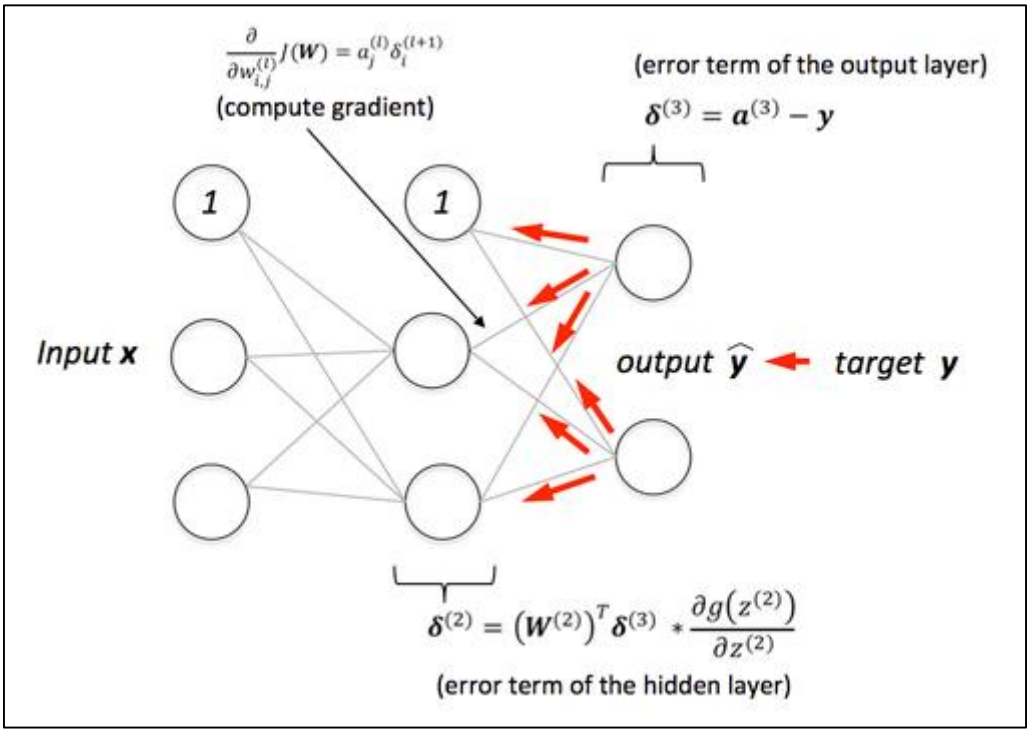


Figure 2.10: Back propagation (propagation and weights update in backpropagation algorithm) [34]

2.5.5 Overfitting

A common issue in machine learning and especially in deep learning is the problem of overfitting [30]. Overfitting is when the model more closely describe the noise in the function than the underlying function, a detailed view is included in Figure 2.11

Overfitting is reduced with a smaller network and a larger training set. It can also be reduced by dropout

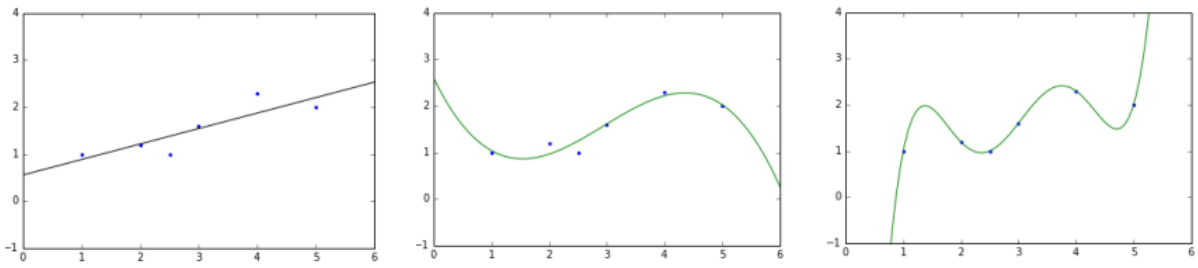


Figure 2.11: Images of different polynomial functions fitted to data (blue points). (Left) An under-fitted function that does not represent the variation of the data. (Middle) A fit that represents the major points that generalize well (Right) A function that resembles the data and the error too closely

2.5.6 Dropout regularization:

Dropout is a regularization technique for neural network models proposed by Srivastava, et al. [35].

Dropout is a technique where randomly selected neurons are ignored during training. They are "dropped-out" randomly. This means that their contribution to the activation of downstream neurons is temporally removed on the forward pass and any weight updates are not applied to the neuron on the backward pass.[36]

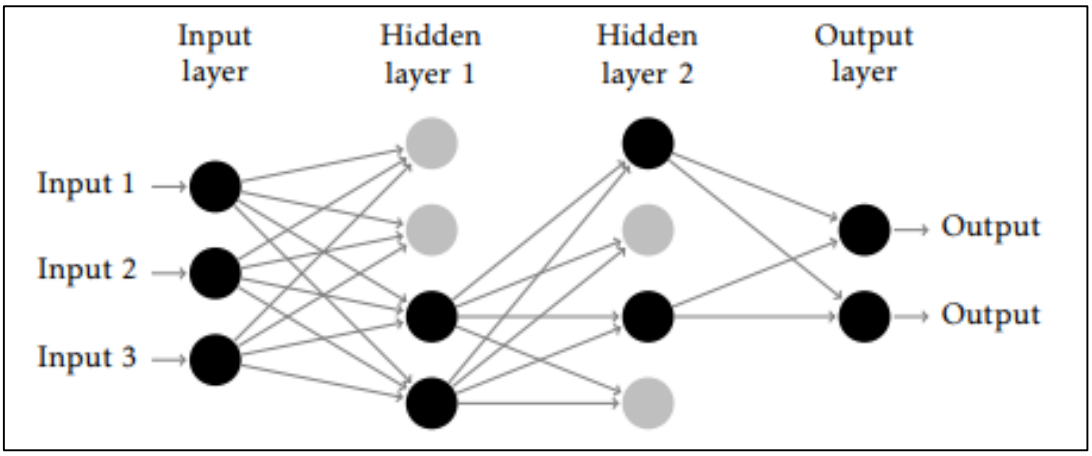


Figure 2.12: A neural network with dropout during training. The activation probability is set to 0:5 and the black nodes are active while the gray are deactivated. The activations are changed between each batch during training and dropout is turned off (all nodes are active) during testing [37].

2.5.7 Data Augmentation

Medical image segmentation is often constrained by the availability of labelled training data. `Data augmentation' helps to prevent memorisation of training data and helps the network's performance on data from outside the training set[38]. As such, it is vital in building robust deep learning pipelines. Augmentation in medical imaging typically involves applying small transformations to training examples, to create variety in the training set[39].

This technique is illustrated in the Figure below, where we show how new training examples for the task of semantic segmentation can be obtained by mirroring, rotating and scaling parts of an image.

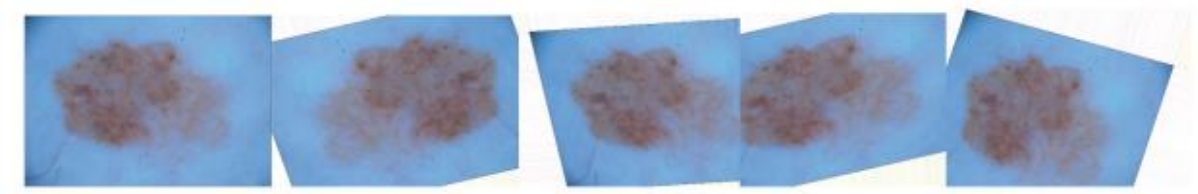


Figure 2.13: Data augmentation

.

2.6 Convolutional Neural Networks

Artificial neural networks have flourished in recent years in the processing of unstructured data, especially images, text, audio, and speech. Convolutional neural networks (CNNs) work best for such unstructured data. Whenever there is a topology associated with the data, convolutional neural networks do a good job of extracting the important features out of the data. From an architectural perspective, CNNs are inspired by multi-layer Perceptrons. By imposing local connectivity constraints between neurons of adjacent layers, CNN exploits local spatial correlation. The core element of convolutional neural networks is the processing of data through the convolution operation. Convolution of any signal with another signal produces a third signal that may reveal more information about the signal than the original signal itself.

2.6.1 CNN Structure:

Convolution neural networks (CNNs) are based on the convolution of images and detect features based on filters that are learned by the CNN through training. For example, we don't apply any filter, such as the ones for the detection of edges or for removing the Gaussian noise, but through the training of the convolutional neural network the algorithm learns image-processing filters on its own that might be very different from normal image-processing filters. For supervised training, the filters are learned in such a way that the overall cost function is reduced as much as possible. Generally, the first convolution layer learns to detect edges, while the second may learn to detect more complex shapes that can be formed by combining different edges, such as circles and rectangles, and so on. The third layer and beyond learn much more complicated features based on the features generated in the previous layer. The good thing about convolutional neural networks is the sparse connectivity that results from weight sharing, which greatly reduces the number of parameters to learn. The same filter can learn to detect the same edge in any given portion of the image through its equivariance property, which is a great property of convolution useful for feature detection[40].

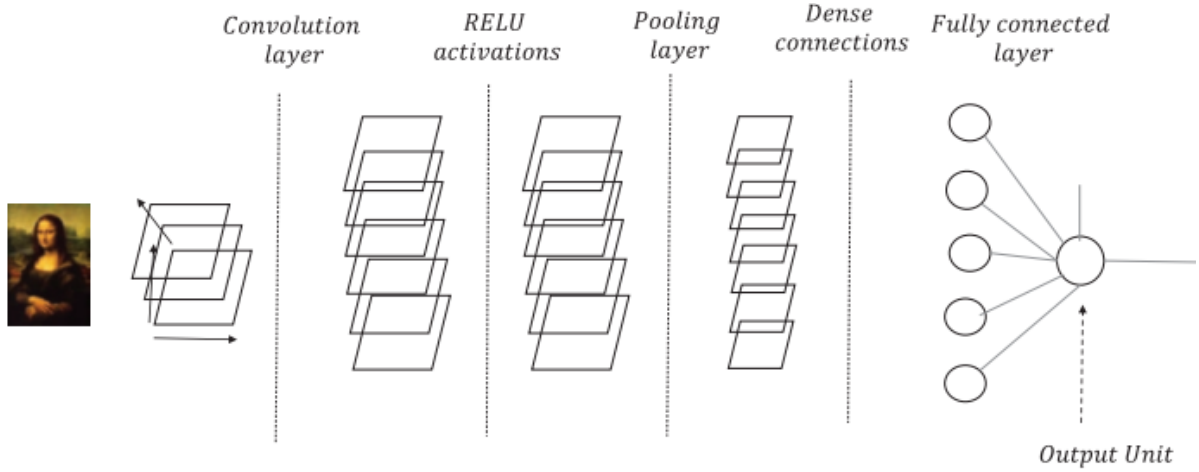


Figure 2.14: Basic flow diagram of a convolutional neural network

The following are the typical components of a convolutional neural network:

Input layer will hold the pixel intensity of the image. For example, an input image with width 64, height 64, and depth 3 for the Red, Green, and Blue color channels (RGB) would have input dimensions of 64 x 64 x 3.

Convolution layer will take images from the preceding layers and convolve with them the specified number of filters to create images called output feature maps. The number of output feature maps is equal to the specified number of filters. Till now, CNNs in TensorFlow have used mostly 2D filters; however, recently 3D convolution filters have been introduced.

Activation function for CNNs are generally ReLUs or Tanh, which we discussed in section 2.4. The output dimension is the same as the input after passing through the activation layers. The activation layer adds non-linearity in the network and at the same time provides non-saturating gradients for positive net inputs.[40]

Pooling layer: A pooling layer operates on blocks of the input feature map and combines the feature activations. This combination operation is defined by a pooling function such as the average or the max function. Similar to the convolution layer, we need to specify the size of the pooled region and the stride. Figure 2.15 shows the max pooling operation, where the maximum activation is chosen from the selected block of values. This window is slided across the input feature maps with a step size defined by the stride (1 in the case of Fig 2.15). If the size of the pooled region is given by $f \times f$, with a stride s , the size of the output feature map is given by[41]:

$$\grave{h} = \left[\frac{h-f+s}{s}\right], \grave{w} = \left[\frac{w-f+s}{s}\right] \qquad (2.12)$$

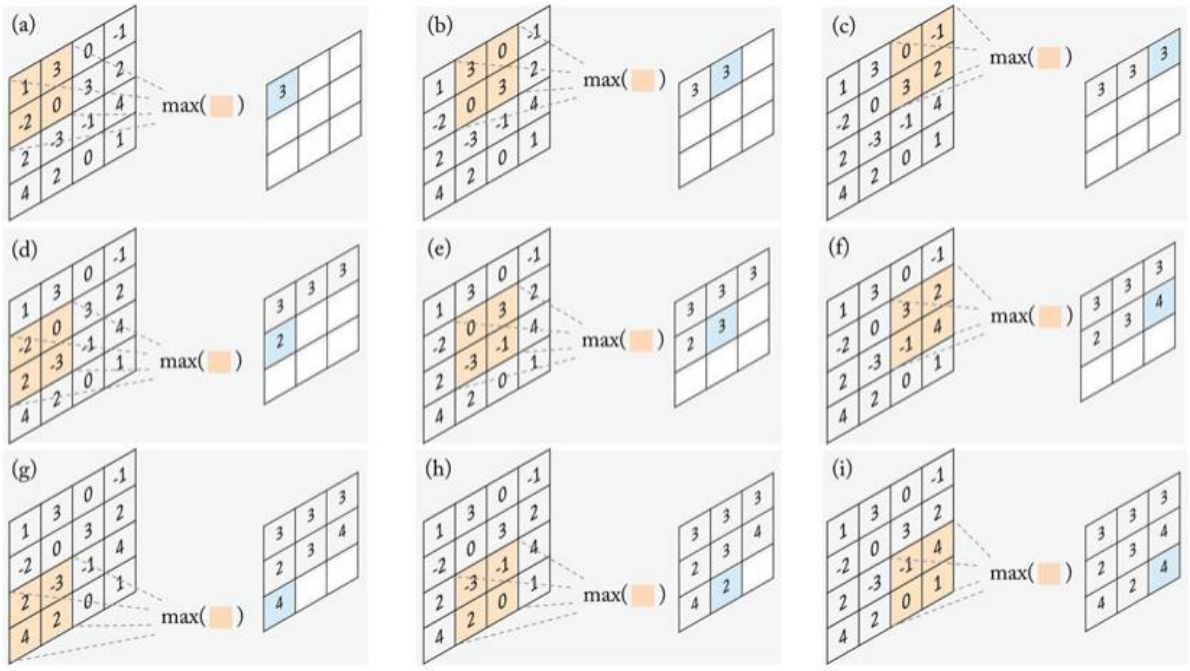


Figure 2.15: The operation of max-pooling layer.

Fully connected Fully connected layers correspond essentially to convolution layers with filters of size 1x1. Each unit in a fully connected layer is densely connected to all the units of the previous layer. In a typical CNN, full-connected layers are usually placed toward the end of the architecture. However, some successful architectures are reported in the literature which use this type of layer at an intermediate location within a CNN. Its operation can be represented as a simple matrix multiplication followed by adding a vector of bias terms and applying an element-wise nonlinear function f:

$$y = f(W^T x + b) \qquad (2.13)$$

Where x and y are the vector of input and output activations, respectively, W denotes the matrix containing the weights of the connections between the layer units, and b represents the bias term vector.[40]

2.7 Transfer learning

Transfer learning in a broad sense refers to storing knowledge gained while solving a problem and using that knowledge for a different problem in a similar domain. Transfer learning has been hugely successful in the field of deep learning for a variety of reasons. Deep-learning models in general have a huge number of parameters because of the nature of the hidden layers and the connectivity scheme within the different units. To train such a huge model, lots of data is required or the model will suffer from overfitting problems. In many problems, the huge amount of data required to train the model is not available but the nature of the problem requires a deep-learning solution in order to have a reasonable impact. For instance, in image processing for object recognition, deep-learning models are known to provide state-of-the-art solutions. In such cases, transfer learning can be used to generate generic features from a pre-trained deep-learning model and then use those features to build a simple model to solve the problem. So, the only parameters for this problem are the ones used to build the simple model. The pre-trained models are generally trained on a huge corpus of data and thus have reliable parameters. When we process images through several layers of convolutions, the initial layers learn to detect very generic features such as curls and edges. As the network grows deeper, the convolutional layers in the deeper layers learn to detect more complex features relevant to the specific kind of dataset. For example, in a classification the deeper layers would learn to detect features such as eyes, nose, face, and so forth.[40]

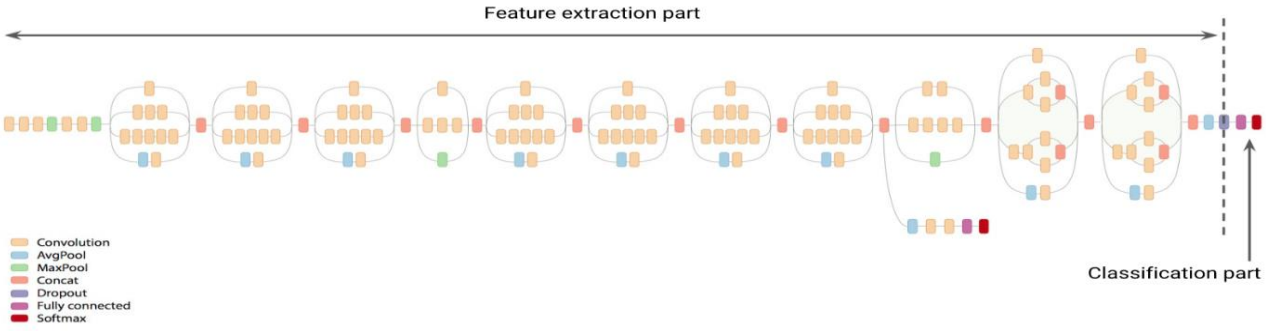


Figure 2.16: image classification transfer learning using Inception-V3

2.7.1 Inception v3:

InceptionV3 is one of the state-of-the-art convolutional neural networks from Google. It's an advanced version of GoogLeNet that won the ImageNet ILSVRC-2014 competition with its out-of-the-box convolutional neural network architecture. The details of the network are documented in the paper titled "Rethinking the Inception Architecture for Computer Vision" by Christian Szegedyand his collaborators.. The core element of GoogLeNet and its modified versions is the introduction of an inception module to do the convolution and pooling. In traditional convolutional neural networks, after a convolution layer we either perform another convolution or max pooling, whereas in the inception module a series of convolutions and max pooling is done in parallel at each layer, and later the feature maps are merged. Also, in each layer convolution is not done with one kernel-filter size but rather with multiple kernel-filter sizes. An inception module is presented in Figure 2.17 below. As we can see, there is a series of convolutions in parallel along with max pooling, and finally all the output feature maps merge in the gilter concatenation block. 1x1 convolutions do a dimensionality reduction and perform an operation like average pooling. [40]

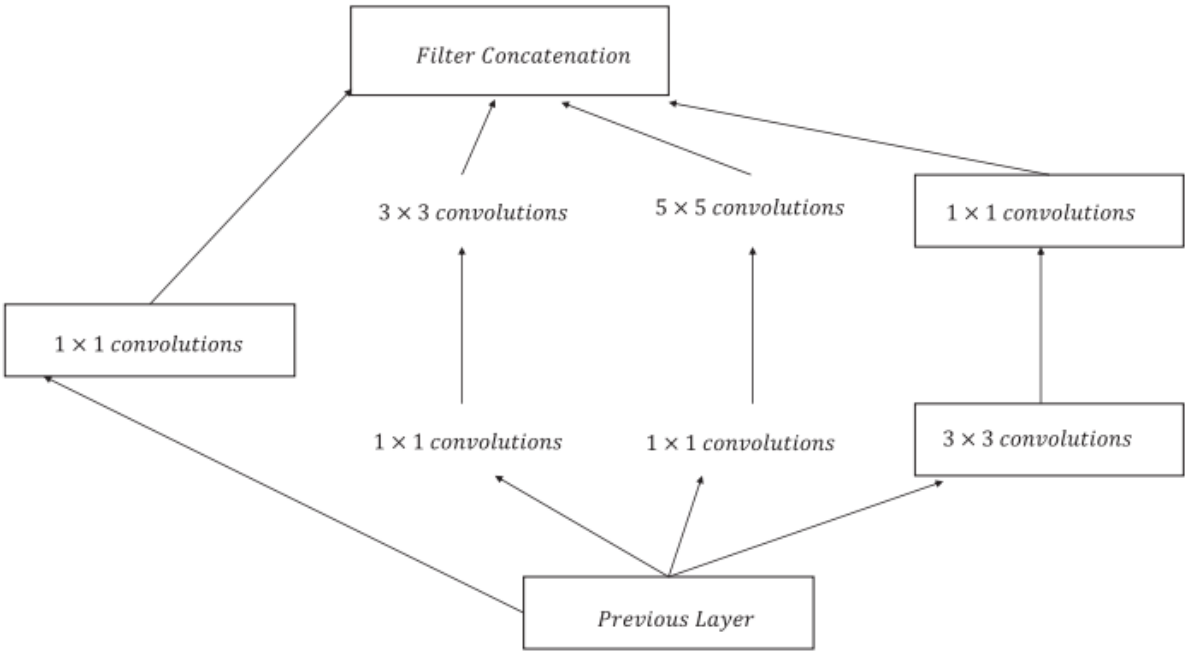


Figure 2.17. Inception module.

2.8: Summary:

In this chapter, we discussed the theory of the artificial neural networks, starting by how it was inspired by the biological neurons, going through also the simple and the multilayer Perceptron, activation functions and training a neural network, before getting around to CNNs and Transfer learning.

CHAPTER 3

Experiments and Results

3.1 Introduction:

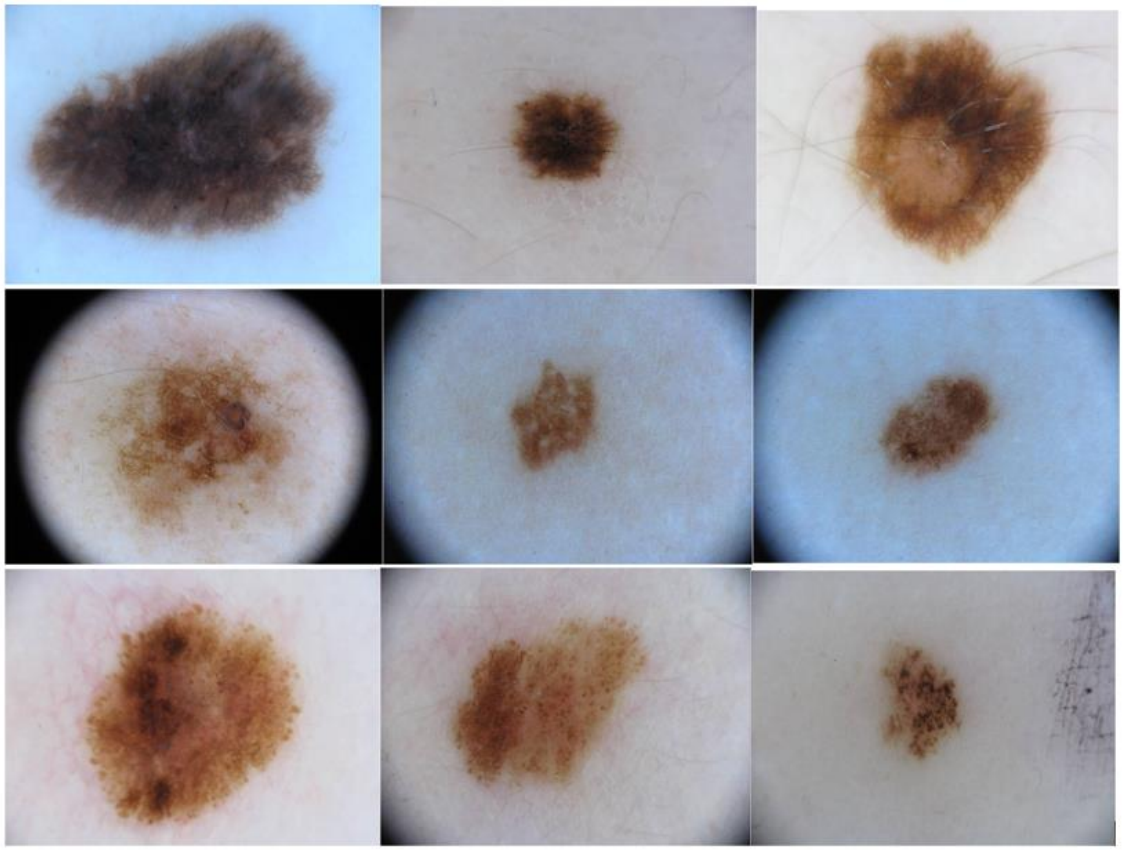
In this chapter, we will present the implementation of Melanoma Identification from Dermoscopy Images using Deep Learning .Considering the importance of choosing the right Data-set and after an intense search, we decided to use the "ISIC" data-set as suggested by state-of-the-art papers. The data is used for training and testing the CNN as described in the following sections.

3.2 ISIC Data-set:

The International Skin Imaging Collaboration (ISIC) dataset currently contains one of the largest collections of contact non-polarized dermoscopy images, complete with manual bounding boxes placed around the lesions for analysis. Examples are shown in Figure.3.1.

The society responsible for the obtaining, organizing and classifying the images of the data-set is The International Society for Digital Imaging of the Skin (ISDIS) . ISDIS was founded in 1992 by a group of U.S. dermatologists, pharmaceutical and cosmetic industry scientists, radiologists, and plastic surgeons. The goal of the society is to familiarize members with new and evolving digital skin imaging technologies such as dermoscopy, telemedicine, magnetic resonance imaging, spectroscopy, optical coherence tomography, total body imaging, etc that are relevant to clinical practice and take steps to advance non-invasive imaging of the skin. ISDIS has been working in close collaboration with International Dermoscopy Society (IDS) and International Confocal Group (ICG) in this regard [42].

A:Benign:



B:Malignant

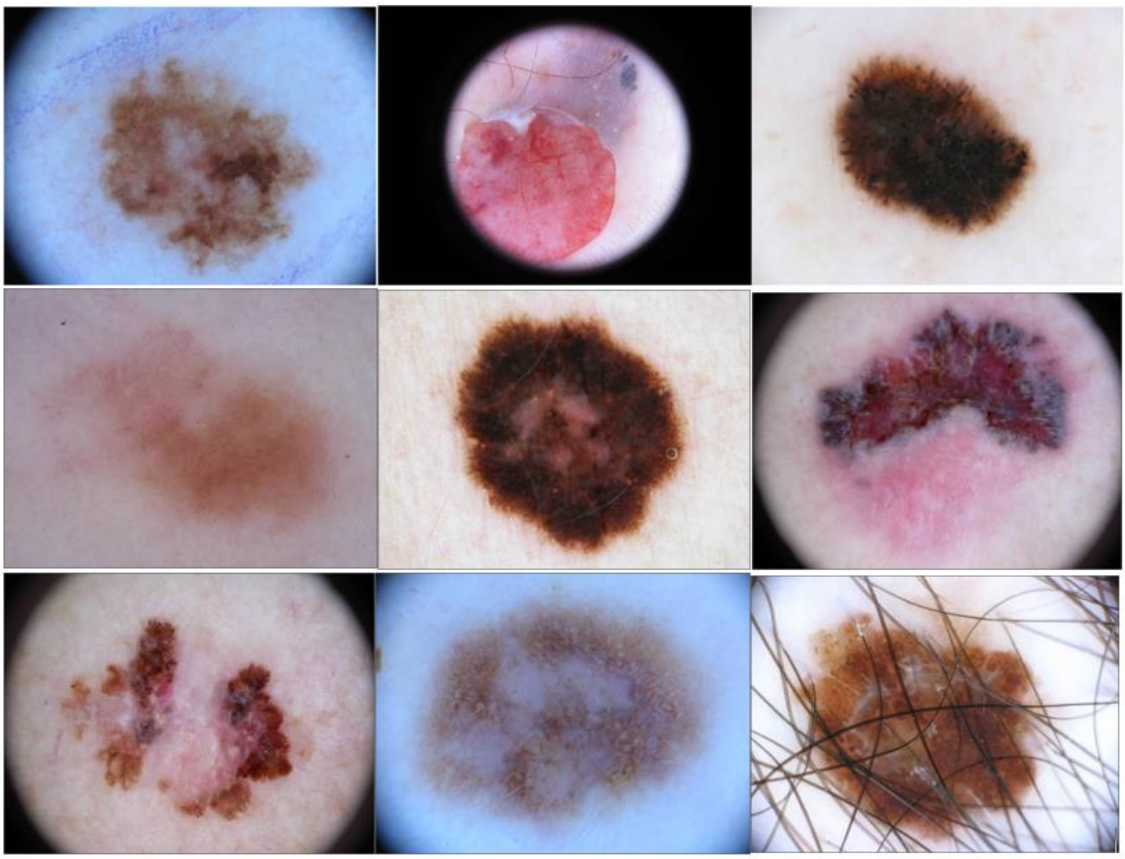


Figure 3.1: *Images from the ISIC Data-set.*

3.2.1 ISIC Data-set image classification:

The following figures represents the classification the images the ISIC Data-set by Benign or Malignant (Figure 3.2), by Age (Figure 3.3) and by Sex (Figure 3.4).

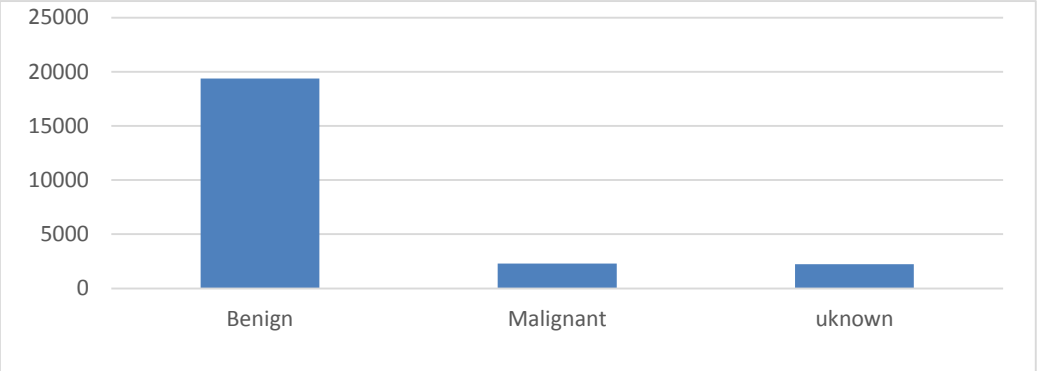


Figure 3.2: *ISIC Data-set image classification by Benign or Malignant.*

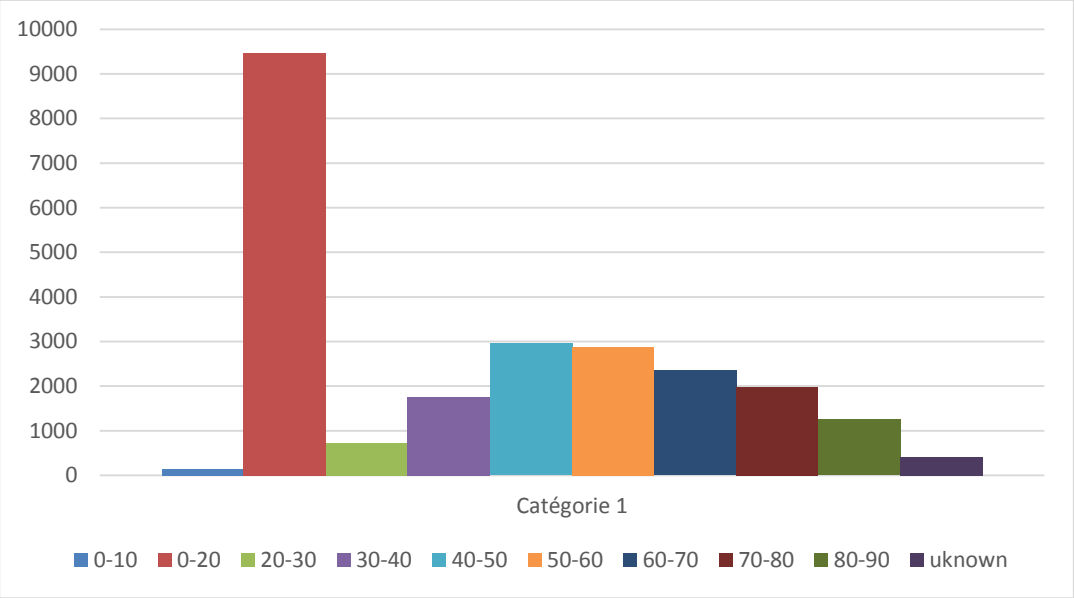


Figure 3.3: *ISIC Data-set image classification by Age*

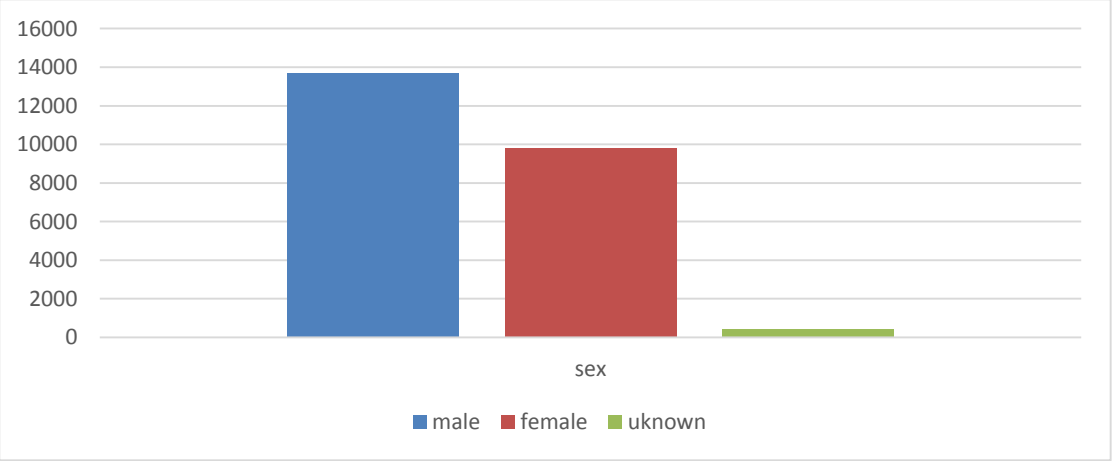


Figure 3.4: *ISIC Data-set image classification by Sex.*

3.3 Training

The data set is split into training, validation and test sets. The training set is used to calculate the loss and to train the network. The validation set is used to evaluate the network after each training phase. It is used to determine when the network has converged. Finally, the test set is used to evaluate the network accuracy.

3.4 Evaluation metrics:

When evaluating clinical tests in medicine there are four common evaluative metrics namely, sensitivity, specificity, positive predictive value (PPV) and negative predictive value (NPV). [42]. the terminology for binary classification is outlined in table 3.1.

Table 3.1: *The terminology for binary classification.*

Classified as Ground truth	Positive	Negative
True	True positive	True negative
False	False Positive	False Negative

3.4.1 Sensitivity & Specificity

Sensitivity and Specificity are two metrics commonly used to evaluate clinical tests; Sensitivity shows how good the test is at detecting a disease. While Specificity suggests how good the test is at identifying normal (negative) condition. Because it is important for a test to find all patients with a certain disease and also to not give treatment to healthy patients [43]. Sensitivity and specificity are calculated according to equation (3.2) and (3.3) respectively.

$$\text{Sensitivity} = \frac{\text{True positive}}{\text{True positive+False Negative}} \tag{3.2}$$

$$\text{Specificity} = \frac{\text{True negative}}{\text{True negative+False Positive}} \tag{3.3}$$

That the maximization of these metrics is often a trade-off which can easily be realized since a "stupid" classifier which classifies everything as Positive will have a Sensitivity of 100%, and similarly a "stupid" classifier classifying everything as Negative will have a Specificity of 100%. Due also to the high rates of classifiers such as these sensitivity and specificity is often accompanied by a predictive value.

3.4.2 Accuracy:

Eq. (3.4) shows the accuracy equation. Accuracy is the proportion of the true diagnostic (negatives + positives) correctly identified by a diagnostic test. It suggests how good the test is at identifying both normal (negative) and melanoma positive conditions.

$$\text{Accuracy} = \frac{\text{True negative+ TruePositive}}{\text{TruePositive+True negative+False Positive+ False negative}} \quad (3.4)$$

3.5 Tools:

The algorithms are written in Python., and the frameworks that were used were Tensorflow and Keras. For the Inception *v3* model because of incompatibility between the versions of the frameworks, the algorithm was trained using a Docker image. Docker is a container technology similar to virtual machines, which consumes less CPU resources and memory. It provides the advantage of independency among different versions of frameworks as well as fast configurations.

3.5.1.TensorFlow

TensorFlow™ is an open source software library for high performance numerical computation. Its flexible architecture allows easy deployment of computation across a variety of platforms (CPUs, GPUs, TPUs), and from desktops to clusters of servers to mobile and edge devices. Originally developed by researchers and engineers from the Google Brain team within Google's AI organization, it comes with strong support for machine learning and deep learning and the flexible numerical computation core is used across many other scientific domains [44].

3.5.2. Keras

Keras is a high-level neural networks library, written in Python and capable of running on top of either TensorFlow or Theano (Keras, 2017). It was developed with the purpose of enabling fast experimentation. Providing results with the least possible delay, Keras constitutes a key for good research. It is designed on the following properties: modularity, so many functions, graphs and neural layers to create new models when combined together,

minimalism, that it is important each module to be short and simple and easy extensibility, the ability to add new models [45].

3.5.3.Python

Python is an open source interpreted high-level programming language for general-purpose programming. Created by Guido van Rossum and first released in 1991, Python has a design philosophy that emphasizes code readability, notably using significant whitespace. It provides constructs that enable clear programming on both small and large scales.

Python features a dynamic type system and automatic memory management. It supports multiple programming paradigms, including object-oriented, imperative, functional and procedural, and has a large and comprehensive standard library.

Python 3.6.0 is the newest major release of the Python language, and it contains many new features and optimizations [46].

3.6 Hyper-parameters

The hyper-parameters used in the algorithms of the classification are learning rate, batch size, epoch and number of hidden layers. These parameters had great influence in the results of the classifier.

Learning rate controls the ability of the algorithm to learn and extracting features from the images during the training. Large-learning rate indicates quick adoption of new features, which can be undesirable for the results, since the algorithm make generalizations easily. Thus, a suitable value for the learning rate should be low enough so as the network to acquire better precision but high enough that it does not require too much time for the training.

Batch size is the number of images that algorithm examines during each training step. Small batch size means that more updates of parameters will happen. However, it may result in lower accuracy, since less images are examined. Again trade-off between the number of updates and the accuracy is needed so as to achieve the best results.

Epoch shows how many times that all the images of the dataset are examined.

Number of hidden layers are the layers between the input and the output layer. Many hidden layers results in better accuracy for the algorithm but also increase the complexity and the training time, and potentially cause over-fitting. The method that determine the most

appropriate hyper-parameters is the Manual Search which can be applied by observing the impact of the hyper-parameters to results and deciding what is best..

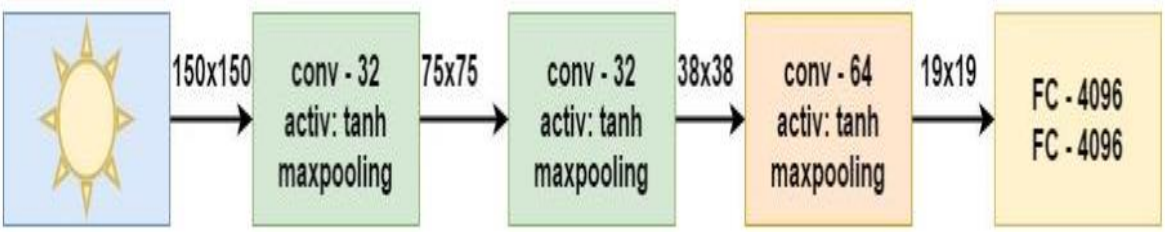
3.7 Experiment and results:

3-7.1 CNN Model:

3-7.1.1 Experiment:

Our first algorithm that we developed a CNN Model using Keras for the training of the dataset. It has three convolutional layers, and for each one we added a "tanh" activation layer and a max-pooling layer (outlined in 2.6). At the end, we added two fully-connected layers as explained in the image (Figure3.5).

Figure 3.5: CNN model Architecture



We resized the dataset images to 150x150 and divided into three categories: a set of 45% for training, a set of 30% for validation, and the remaining 25% for testing. Also, because our dataset is small, and in order to make the most of our few training examples and increase the accuracy of the model we used data augmentation (outlined in 2.5). Furthermore, it is expected that data augmentation should also help prevent overfitting (a common problem in machine learning related to small datasets). Rescaling, shearing, zooming and horizontal flipping were implemented to the dataset.

3.7.1.2 Results:

The results of both experiments (with and without using data augmentation) that are shown in Tables (3.2), knowing that we have used 50 epochs and batch size equal to 16. The total time for every training was estimated around 9 hours.

Table 3.2: *Training results using images with/without data augmentation*

DATASET	N° of Cases	Sens (%)	Spec (%)	Acc (%)
ISIC Archive With Data augmentation	**1500**	**87.04**	**42.9**	**60.67**
ISIC Archive Without Data augmentation	**1500**	**0.45**	**99.16**	**55.53**

3.7.2 Transfer learning

3.7.2 .1 Experiment

Inception *v3* is a network trained on ImageNet database designed for image recognition tasks. It includes 174 layers, which are stacked on top of each other. As required by the model, the images were resized to 299x299 and placed in two separated folders, melanomas and non melanomas.

Firstly, the pretrained model of Inception *v3* is loaded and then bottlenecks (activation maps before the fully-connected layers). The images were randomly divided into 60% for training, 20% for validation and 20% for testing. During the training, the 2 final fully-connected layers of the network are removed and the new one is added in order to train the images of our project. In every step of the training, according to the batch size, a specific number of images is examined and validation accuracy as well as validation loss are calculated. In the end, this process is repeated once more but only for the test images providing the final results. Even though ImageNet includes around 2 million images, it does not include images of melanomas and non melanomas but many features are extracted from the other images that assist in the classification of any type of images.

3.7.2 .1 Results

The sensitivity, specificity and accuracy results with regard to melanoma for Inception architecture are collected and shown in Table 3.3:

Table 3.3: *Training results for Inception v3 Model fine tuning.*

Dataset	No of Cases	Sens (%)	Spec (%)	Acc (%)
ISIC Archive	1500	72.08	90.91	83.7

In Table 3.4, the dataset of ISIC, was used so as to show how the different values of hyper-parameters (learning rate and batch size) affect the sensitivity, specificity and accuracy.

The batch size has been fixed at 200 and we varied the learning rate.

Table 3.4: *Training results for different values of learning rate.*

Evaluation Metrics learning rate	Sensitivity(%)	Specificity(%)	Accuracy (%)
0.1	79.59	95.10	90.23
0.01	**87.75**	**92.27**	**91.41**
0.001	72.08	90.91	83.7
0.0001	89.79	79.22	81.25

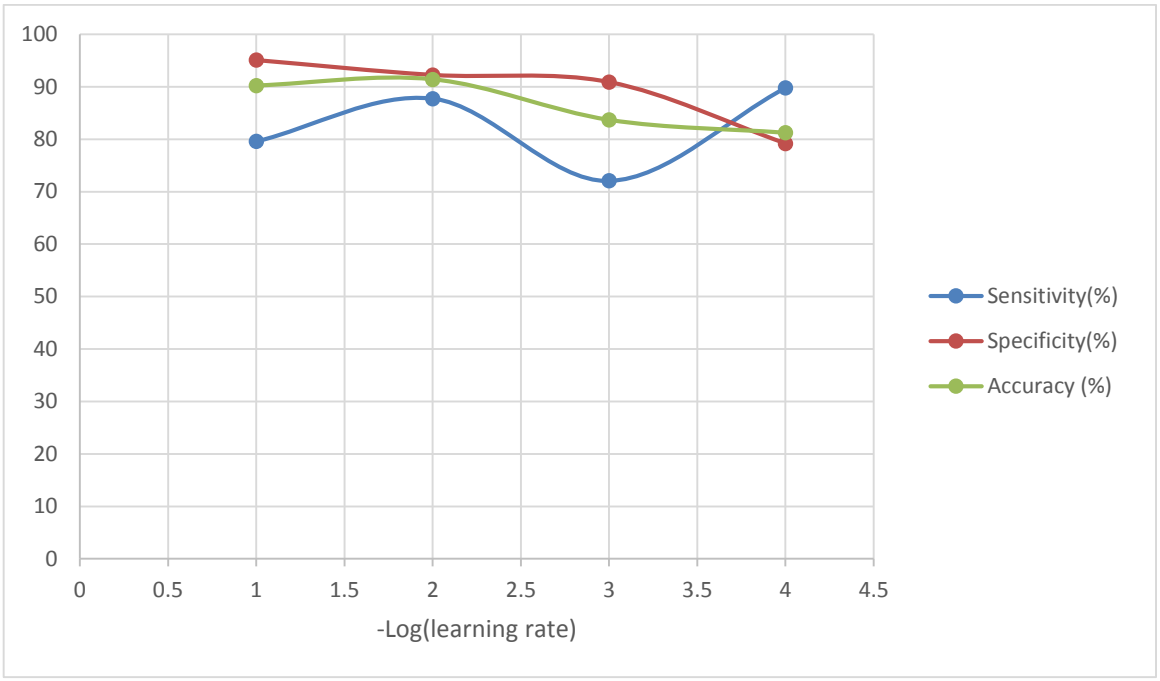


Figure 3.6 : *Classification accuracy, Specificity and Sensitivity with different values of learning rate.*

Since a 0.01 learning rate gave us the best results, the learning rate has been fixed at that value and the batch size have been varied. Results are shown in Table 3.5 .

Table 3.5: Training results for different values of batch size.

Evaluation Metrics Batch size	Sensitivity(%)	Specificity(%)	Accuracy (%)
10	82.43	93.79	87.15
100	85.36	93.15	89.78
200	**87.75**	**92.27**	**91.41**
500	83.67	92.27	90.63

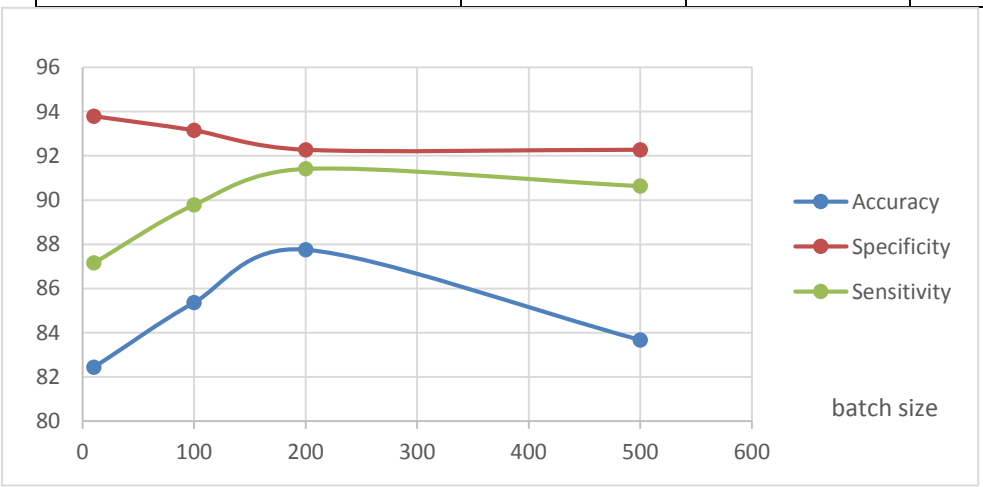


Figure 3.7 : *Classification accuracy, Specificity and Sensitivity with different values of batch size.*

3.8 Discussion:

Deep learning algorithms were implemented with the purpose of classifying the images of the datasets into melanoma and non-melanomas. The algorithm was used as a reference point, and since the number of layers was low the results were not perfect.

Also, the images were classified since they underwent with and without data augmentation in. As it can be seen on Tables 3.2, the algorithm without data augmentation did not perform well with the images. In all cases, all the images were classified as melanomas or non-melanomas so specificity and sensitivity were 0.45% and 99.16% respectively. However when the data augmentation were applied the results were more accurate, providing 87.04%, accuracy, 42.9% sensitivity and 60.67% specificity .

The method with the best performance for dermoscopical images was proved to be the one using Transfer Learning with the architecture of Inception v3. It achieved accuracy 83.7%, sensitivity 74.68% and specificity 90.91%.

In order to obtain more accurate results , different values for learning rate and batch size were used so as to compare the results of the algorithm. We tried to find the optimal hyper-parameters(learning rate and the Batch size). First, the batch size has been fixed at 200 and the learning rate has been varied to 0.1, 0.01, 0.001 and 0.0001.Next, the learning rate has been fixed at 200, and the batch size have been varied (10,100,200,500).

It is observed that with high learning rate, specificity increases but with trade-off of the sensitivity. Which is not so useful since many melanomas are classified as non-melanomas, Which can be really misguided. However, For a small value of learning rate, accuracy and specificity are quite degraded. For the batch size, very small values give high specificity but low sensitivity, and when increasing the batch size, the specificity reduces and the sensitivity increases. This is due to the fact that more images are taken into consideration in every step of the process and our algorithm adopts many features also from images of melanomas. When we use small training batch size, features extraction is from nevi images since they are more in number which results in high specificity. It is found that the optimal values for the hyper-parameters for this model is 0.01 and 200 for the learning rate and the batch size, respectively. For deep learning, bigger datasets are required to achieve more accurate results, and for medical images, not many are available. But by creating the appropriate architecture of deep learning model as well as good hyper parameters can provide the best results in addition to data augmentation techniques, as it was shown in this project.

The rate of clinician's visual investigations gets a moderate detection as shown in **Table 3.6**. . It is very difficult to distinguish some atypical lesions - which are benign - from melanoma because they have the same properties according to the well-known rules used by dermatologists.

		Sensitivity(%)	Specificity(%)
Argenziano et al. (2006)	All malignant	**79.2**	**71.8**
Menzies et al. (2009)	**Only melanoma**	53.1	89.0
This work (2017)	Only melanoma	72.08	83.7

Table 3.6. Comparison the sensitivity and specificity of this work with clinicians

The present work is proposed as an additional tool for not experienced dermatologists in order to strengthen the confidence of their results or as private use to check the evolution of skin lesions in the body.

3.8 Future work:

Further research can be focused on improving Accuracy, sensitivity and the specificity, there are multiple ways to achieve that is to use the latest trend in Deep learning, which is "Super deep learning".

After training a an already existing models (like the one we build from scratch , Inception *v3*, VGG-16…) to recognize and classify Melanoma pictures, a super learner is build that's take the results of the pre-trained Models as inputs to increase the Accuracy. Like it is shown in figure 3.7.

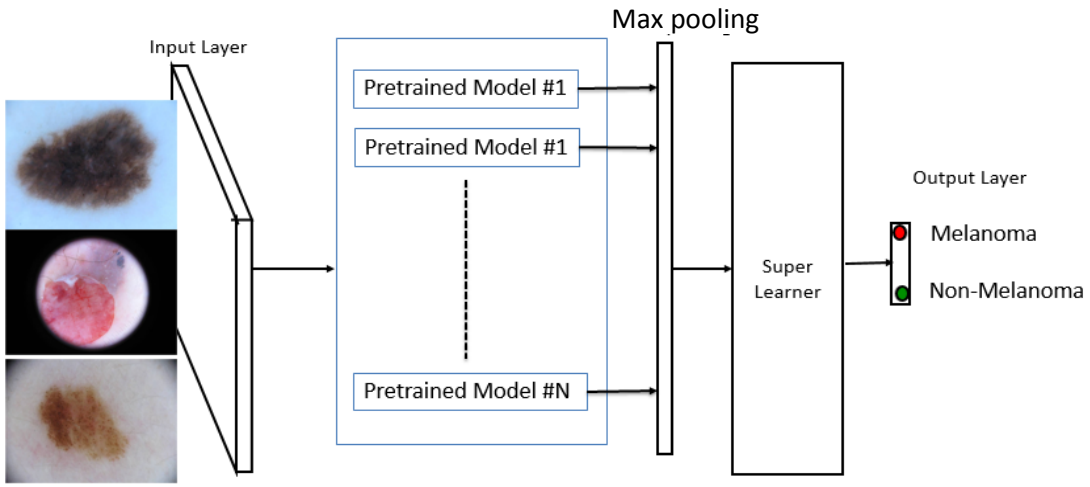


Figure 3.7: *Super deep learner architecture.*

3.8. Summary

The performance of convolutional neural networks model built from scratch is found to be unsatisfactory compared to transfer learning method where we used Google's Inception (pre-trained on other datasets with more or less related data).

Convolutional neural networks present an opportunity for computer assisted Melanoma Identification from Dermoscopy Images.

CONCLUSION

Conclusion

The aim of the project was to classify, with the use of Deep Learning, dermoscopy images of benign–melanoma. The performance of convolutional neural networks is found to be comparable to practicing dermatologists and state of the art machine learning methods in binary classification accuracy, with only little pre-processing and tuning (Inception *v3*).
Augmentation in medical imaging proved to be important, especially when training datasets are small, it helps improving accuracy and also reduce overfitting.

The method with the best performance for dermoscopical images was proved to be the one using Transfer Learning with the architecture of Inception v3. It achieved accuracy 83.7%, sensitivity 74.68% and specificity 90.91%.

Different values for learning rate and batch size were used so as to compare the results of the algorithm. We tried to find the optimal hyper-parameters(learning rate and the Batch size).

In these last years, Computer assisted diagnosis have been a lively field of research. Many algorithms have been developed as a tool in order to help clinicians or even as a primary opinion for suspicious individual users. Many approaches were used (SVM, Very Deep Learning) to tackle this problem, Achieving moderate to satisfactory results.

Convolutional neural networks present an opportunity for decision support in skin lesion screening. So, an application using the present technique can provide reliable results and serve as an assisting tool from dermatologists and self-examination tool for first diagnosis from individual users.

BIBLIOGRAPHY

BIBLIOGRAPHY

[1] "Melanoma Treatment–for health professionals (PDQ®)". National Cancer Institute. June 26, 2015

[2] https://www.cancer.org/cancer/melanoma-skin-cancer . American cancer society.

[3] NCI's PDQ cancer information summary, 2016

[4] Tsao H. et al., 2012

[5] World Cancer Report 2014 (PDF). World Health Organization. 2014. pp. Chapter 5.14

[6] GBD 2015 Disease and Injury Incidence and Prevalence, Collaborators. (8 October 2016). "Global, regional, and national incidence, prevalence, and years lived with disability for 310 diseases and injuries, 1990–2015: a systematic analysis for the Global Burden of Disease Study 2015"

[7] Oliveria S, Saraiya M, Geller A, Heneghan M, Jorgensen C (2006). "Sun exposure and risk of melanoma"

[8] Rünger TM, Farahvash B, Hatvani Z, Rees A (January 2012). "Comparison of DNA damage responses following equimutagenic doses of UVA and UVB: a less effective cell cycle arrest with UVA may render UVA-induced pyrimidine dimers more mutagenic than UVB-induced ones"

[9] Rhodes A, Weinstock M, Fitzpatrick T, Mihm M, Sober A (1987). "Risk factors for cutaneous melanoma. A practical method of recognizing predisposed individuals"

[10] "MelanomaWarningSigns.com"

[11] Fiddler IJ (October 1995). "Melanoma metastasis". Cancer Control. 2 (5): 398–404

[12] Balch C, Buzaid A, Soong S, Atkins M, Cascinelli N, Coit D, Fleming I, Gershenwald J, Houghton A, Kirkwood J, McMasters K, Mihm M, Morton D, Reintgen D, Ross M, Sober A, Thompson J, Thompson J (2001). "Final version of the American Joint Committee on Cancer staging system for cutaneous melanoma"

[13] Friedman R, Rigel D, Kopf A (1985). "Early detection of malignant melanoma: the role of physician examination and self-examination of the skin"

[14] W. Stolz, A. Riemann, A. B., A. Cognetta, L. Pillet, W. Abmayr and e. al., "ABCD rule of dermatoscopy: a new practical method for early recognition of malignant melanoma," European Journal of Dermatology, vol. 7, no. 4, 1994.

[15] Argenziano G, Fabbrocini G, Carli P, De Giorgi V, Sammarco E, Delfino M. Epiluminescence microscopy for the diagnosis of doubtful melanocytic skin lesions: comparison of the ABCD rule of dermatoscopy and a new 7-point checklist based on pattern analysis. Arch Dermatol 1998; 134: 1563–1570

BIBLIOGRAPHY

[16] Argenziano et al.: Dermoscopy of pigmented skin lesions: results of a consensus meeting via the Internet. J. Am. Acad. Dermatol. 2003;48:679-93. PMID: 12734496. DOI.

[17] Menzies et al.: Frequency and morphologic characteristics of invasive melanomas lacking specific surface microscopic features. Arch Dermatol1996;132:1178-82. PMID: 8859028.

[18] H. Pehamberger, A. Steiner and K. Wolff, "In vivo epiluminscence microscopy of pigmented skin lesions. I. Pattern analysis of pigmented skin lesions," Journal of the American Academy of Dermatology, vol. 17, no. 4, pp. 571-583, 1987

[19] H. Kittler, C. Rosendahl, A. Cameron and P. Tschandl, Dermatoscopy: An algorithmic method based on pattern analysis, Vienna: Facultas Verlags- und Buchhandels AG, 2011.

[20] http://www.skincarenetwork.co.uk/

[21] https://www.curemelanoma.org/patient-eng/melanoma-treatment/

[22] http://www.skincarenetwork.co.uk/skin-cancer-treatment/dermoscopy/

[23] Rojas, R. Neural Networks - A Systematic Introduction. SpringerVerlag, Berlin, New-York, 1996.

[24] Long, L. N., and Gupta, A. Scalable massively parallel artificial neural networks. Journal of Aerospace Computing, Information, and Communication 5, 1 (2008).

[25] Artur Vitt, Mauro Caporuscio Machine Learning in Digital Telerehabilitation,2015

[26] Anna Gummeson,Prostate Cancer Classification using Convolutional Neural Networks,2015.

[27] X. Glorot, A. Bordes, and Y. Bengio, \Deep sparse rectifier neural networks," in *JMLR W&CP: Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics (AISTATS 2011),* Apr. 2011.

[28] A. L. Maas, A. Y. Hannun, and A. Y. Ng, \Rectifier nonlinearities improve neural network acoustic models," *Proc. ICML,* vol. 30, 2013.

[29] https://www.saylor.org/site/wp-content/uploads/2011/11/CS405-6.2.1.2-WIKIPEDIA.pdf

[30] M. Nielsen, Neural Networks and Deep Learning, Determination Press, 2015.

[31] I. Goodfellow, Y. Bengio and A. Courville, Deep learning, Cambridge: Unpublished (MIT Press), 2016.

[32] Goodfellow, I., Bengio, Y., and Courville, A. *Deep Learning*. MIT Press, 2016.

BIBLIOGRAPHY

[33] Wilson, D. R., and Martinez, T. R. The general inefficiency of batch training for gradient descent learning. *Neural Networks 16*, 10 (2003), 1429{1451.

[34] https://github.com/rasbt/python-machine-learning-book/blob/master/faq/visual-backpropagation.md

[35] Dropout: A simple way to prevent neural networks from overfitting. N Srivastava, G Hinton, A Krizhevsky, I Sutskever, R Salakhutdinov. The Journal of Machine Learning Research 15 (1), 1929-1958

[36] https://machinelearningmastery.com/dropout-regularization-deep-learning-models-keras/

[37] Face Recognition with Preprocessing and Neural Networks. David Habrman. Linkoping University

[38] Improving Data Augmentation for Medical Image Segmentation. *Zach Eaton-Rosen, Felix Bragman, Sebastien Ourselin, M. Jorge Cardoso*

[39] https://medium.com/nanonets/how-to-use-deep-learning-when-you-have-limited-data-part-2-data-augmentation-c26971dc8ced

[40] Pro Deep Learning with TensorFlow, Santanu Pattanayak,2017

[41] A Guide to Convolutional Neural Networks for Computer Vision. Salman Khan, Hossein Rahmani, Syed Afaq Ali Shah ,Mohammed Bennamoun,2018.

[42] https://isdis.net/about/

[43] A. G. Lalkhen and A. McCluskey, "Clinical tests: sensitivity and specificity," Continuing education in anaesthesia, critical care & pain, vol. 8, no. 6, pp. 221-223, 2008.

[44] https://www.tensorflow.org/

[45] https://keras.io/

[46] https://www.python.org/downloads/release/python-360/