Master's thesis

2021

Maël Nedellec, Quentin Mouret

Master's thesis

NTNU
Norwegian University of
Science and Technology
Faculty of Information Technology and Electrical
Engineering
Department of Information Security and Communication
Technology

Maël Nedellec
Quentin Mouret

Deep Learning Algorithms in Health area: Adversarial Attacks and Countermeasures

June 2021

NTNU
Norwegian University of
Science and Technology

NTNU
Norwegian University of
Science and Technology

NTNU
Norwegian University of
Science and Technology

Deep Learning Algorithms in Health area: Adversarial Attacks and Countermeasures

Maël Nedellec

Quentin Mouret

Master of Science Programme in Communication Technology, Information Security
Submission date: June 2021
Supervisor: Sule Yildirim Yayilgan

Norwegian University of Science and Technology
Department of Information Security and Communication Technology

Title: Deep Learning Algorithms in Health Area: Adversarial Attacks and Countermeasures

Student: Maël Nedellec & Quentin Mouret

Problem description:

Nowadays, Machine learning is a quite common tool used to assist human in the decision making. More precisely, in the Health area deep learning algorithms are used to help the medical corps in the diagnosis and the prognosis processes.

New techniques always bring new vulnerabilities and in the health area a need to secure these algorithms is more than essential, it could be life critical.

Our work will focus on finding a way to improve the performance of some already implemented countermeasures, but also, would it possible to transfer countermeasures used on other datasets, area or even models to our problem: Deep learning in health Area.

Date approved: 2021-02-17

Supervisor: Sule Yildirim Yayilgan, IIK

Abstract

Nowadays, in the health area, Artificial Intelligence (AI) becomes a must-have to improve diagnosis and prognosis quality. Thus, the medical corps can use Deep Learning (DL) algorithms to predict the evolution of diseases, such as breast or skin cancers, and also detect diseases using medical image analysis. As it can mimic human work – and sometimes, performs better work – it is a powerful tool that can save lives. However, as soon as we talk about algorithms, we have to talk about possible adversarial attacks. Since algorithms handle health data, if an attack makes it badly trained, it could become life-critical.

Our thesis motivation is to investigate the behaviour of such methods in a health-oriented classification model and the potential effectiveness of combining several countermeasures to mitigate these adversarial attacks.

In the health area, DL is used both in prognosis, to predict the development of a disease such as colon, breast, or skin cancer, and in diagnosis to detect and prevent disease. Medical image analysis using AI techniques to extract information from medical images, and may be combined with a classification model, for instance, by using Convolutional Neural Network (CNN) for melanoma classification.

Several attack types exist in the literature. Firstly, Fast Gradient Sign Method (FGSM) and Universal Adversarial Perturbation (UAP) are evasion attacks, as well as the attack proposed by Carlini & Wagner (REF). There are also poisoning attacks, that add skewed data to the training dataset. To counter these attacks, there are three types of countermeasures. We can modify the model to add robustness (Defensive Distillation, Gradient Regularisation), or alter the dataset (Low-Level Transformation, Adversarial Retraining, Online Alternate Generator), or finally using an additional model (Generative Adversarial Network).

We have performed experiments on two neural networks, Residual Network (ResNet50) and Inception V3. As there were several experiments, we chose to focus on only one dataset, ISIC skin lesion 2018, composed of 7 classes (4 cancerous (86% of the pictures), 3 benigns). We picked two evasion attacks, FGSM for its ease of implementation and its impact on the DL classifier, and UAP, for being a recent and powerful attack. Concerning the countermeasures, we wanted to use a less complex method. To investigate if we could mitigate powerful attacks with such countermeasures.

All this work has been performed under python, using Keras and Tensorflow libraries, to answer three questions. Firstly, "how would the classifiers - combination of DL model and ISIC2018 dataset - would be impacted by adversarial attacks?" Second question is, "Is there a way to mitigate FGSM attack?", and the third one is "Can we get the same results against UAP with these methods or by combining them?" To evaluate the results, we use different metrics, such as accuracy, recall, and specificity. Furthermore, we will focus on the False Negative Rate (FNR), which points to the percentage of sick patients classified as healthy (and that can be life-critical in case of skin cancer).

There have been a total of 3 main experiments. First, Inception V3 and ResNet50 have been implemented and evaluated, with above 90% accuracy and between 5 and 8% FNR. In the second experiment, we have performed attacks on the models. After FGSM, we obtained 40% accuracy & 57% FNR for Inception V3, and 60% accuracy & 34% FNR for ResNet50. After UAP, we obtained 37% accuracy & 69% FNR for Inception V3, and 64% accuracy & 28% FNR for ResNet50.
The third step was to mitigate these attacks, with Adversarial Retraining, LLT, and a combination of both. For all models and attacks, the association of both countermeasures has given the best results. In terms of results, Inception V3 and ResNet50 ended with around 85% of accuracy and a low false-negative rate around 7% under both FGSM and UAP attacks.

Whether under FGSM and UAP attack, both ResNet50 and Inception V3 models got unacceptable results according to the metrics. However, our experiments show that these attacks can be mitigated, and so allow to use of these models in the health area. Nevertheless, and as long as model FNR is not down to 0%, it seems important to continue to double-check the results after model predictions. Even if our results are good, we have thought of several complementary experiments for future work. We would recommend, at least, experiment with these methods with other datasets and models.

Acknowledgements

We would like to thank our thesis supervisor, Associate Professor Dr Sule Yildirim Yayilgan. We are gratefully indebted to her for all her valuable comments on this thesis. During the whole thesis, she kept proposing papers and conferences, to give us as much knowledge as possible. She kindly proposed weekly meetings, during which she checked our good progress and steered us in the right direction whenever she thought we needed it.

Preface

This master thesis report is written to meet the requirements for obtaining a diploma from both the "Norwegian University of Science and Technology - NTNU" of Trondheim, Norway, and the "Institut National des Sciences Appliqués - INSA", of Toulouse, France. The thesis *Deep Learning Algorithms in Health Area: Adversarial Attacks and Countermeasures* is linked to a preliminary state of the art we wrote during the autumn semester 2020 and named *Adversarial Attacks Against Deep Learning Algorithms and Mitigation Methods*[QM20].

Contents

Chapter 1 Introduction

1.1 Lexicon

IA: Artificial Intelligence **ML**: Machine Learning **DL**: Deep Learning
CNN: Convolutional Neural Network **DNN**: Deep Neural Network

FGSM: Fast Gradient Sign Method **UAP**: Universal Adversarial Perturbation
RT: Adversarial Retraining **LLT**: Low Level Transformation

TP: True Positive **FP**: False Positive **TN**: True Negative **FN**: False Negative
NPV: Negative Predictive Value **FNR**: False Negative Rate

1.2 Keywords

machine Learning, deep Learning, adversarial attacks, medical field, mitigation methods, ISIC skin lesion dataset

1.3 Justification, Motivation, and Benefits

As mentioned previously, in the health domain, we consider machine learning algorithms and more precisely deep learning with Deep Neural Networks (DNN) a very interesting tool to assist the medical corps during the decision-making process.

Researches have reported several benefits regarding the use of DNN in the literature in various domains and more precisely, in the health area. Among them we could mention classification models to detect several diseases through image classification, increasing the chance of recovery. However, new techniques and methods to help human in any process brings new vulnerability and machine learning models are continuously targeted and subjected to adversarial attacks, thus, the need to secure these systems is more and more effective as it may be life-critical. Hence, researchers have presented several methods to secure Machine Learning (ML) model several are mitigating adversarial attacks at a high percentage. However, it is sometimes performed on "simpler" models and datasets, it is interesting to transfer different methods, such as an online alternate generator or the use of low-level transformation

It motivates to investigate the behaviour of such methods in a health-oriented classification model and the potential effectiveness of combining several countermeasures to mitigate these adversarial attacks. Moreover, as mentioned previously, DL, especially in the medical field is even more vulnerable to attacks. Since researchers have discovered a new form of adversarial attacks called Universal Adversarial Perturbation (UAP) and have proven that health-oriented models are vulnerable to this kind of attack. This gives the motivation to find a way to mitigate such powerful attacks, and it would benefit everyone for the good of medical progress.

1.4 Research Questions

As mentioned previously, while current research on adversarial attacks and countermeasures are often oriented to non-medical datasets and models. There is a need to also look into adversarial attacks and countermeasures in the medical domain. Several methods have been described and experimented with an online alternate generator or low-level transformation as countermeasures to adversarial attacks. It would be beneficial to transfer and utilise such techniques in the health domain and search for any improvement in these methods in terms of performances. Hence, alongside the discovery of some UAPs, it would be medically important to see how we simulate (apply these perturbations on the medical datasets) and mitigate such perturbations (attacks) in the health domain. RESEARCH GOAL: We aim to transfer such methods and we would like to improve the performances. Consequently, we have established the following research questions:

RQ1: How would the classifiers - combination of DL model and ISIC2018 dataset - be impacted by adversarial attacks (e.g. FGSM, UAP)?

RQ2: Concerning adversarial attacks, more precisely for FGSM, is there a way to mitigate this attack on this dataset?

RQ3: Is there a way to achieve the same results against UAP (Presented by Hirano et. al.[HMT20]) by using simplified methods or by combining known methods (e.g. Adversarial training and Low level Transformation)?

1.5 Thesis Structure

The thesis structure aim to give an answer to each research question.

1. **Introduction**
   In this chapter, an introduction to the topic, problem description and the related research questions are discussed

2. **Related Work**
   In this chapter, a literature review is done concerning the research questions we had in mind. The main goal of this chapter is to provide the required background to well understand the topic.

3. **Methodology**
   In this chapter, the way we worked to answer the research questions is detailed as well as the various experiments we have conducted.

4. **Experiments**
   In this chapter, all the experiments results are presented and explained.

5. **Discussion**
   This chapter contains further analysis of the results for each experiment and answers to the research questions.

6. **Conclusion and Future Work**
   This chapter closes the Master Thesis, summarising all previous chapters and proposing possible future works.

Chapter 2
Related Work

2.1 Deep Learning in the Health Area

The state of art and related work were reviewed, and an identification of the relevant background material was carried out in the project preceding this thesis[QM20]. This is amended with a discussion of a few papers that have been studied after the project.

The use of ML and DL in the medical field is various. There are four major applications in which DL methods are beneficial; in this thesis we have focused on the prognosis and diagnosis process.

- The prognosis, in clinical processes, means to predict the expected development of a disease. In this process, the goal is to identify symptoms of a specific disease to know whether they will improve or on the contrary, become worse. For instance, ML models have been developed to predict colon cancer [LDJ+20] or also breast cancer [FZR+19]. These two examples aim at the same thing: personalised medicine through individual prognostics. Moreover, due to the ongoing coronavirus pandemic, a 3D - Convolutional Neural Network (CNN) has been developed to help alert for COVID-19 patients at High-Risk of death [MDL+20]. In the prognosis, the medical corps can use ML algorithms or models during the disease.
- On the contrary, for the diagnosis process, health professionals use ML to prevent complication of diseases with an early diagnosis. There are several applications in real situations, one of the most important is medical image analysis. Medical image analysis uses Artificial Intelligence (AI) techniques to extract information from medical images acquired by different systems (e.g. magnetic resonance imaging (MRI), ultrasound, X-ray). The main goal of medical image analysis is to assist the medical corps in decision making to give the best diagnostic and prognostic of a specific disease. Methods used to reach this goal can be "enhancement", for instance, a model enhance medical images to reduce noises or disturbances encountered during the acquisition [Gon16].

> More related to our project, researchers have used ML as a detection model, to identify specific abnormalities in medical images, such as tumour or cancer. Moreover, it may be combined with a classification model, for example, using CNN for melanoma (a form of skin cancer) classification [DYYH19, MYYH16].

As mentioned, the use of ML/DL for healthcare is various. Hence, there are several datasets, and each dataset corresponds to a specific disease or specialism. We can organise the most commonly used dataset in 3 different types of medical images: chest X-ray images used for pneumonia classification, Optical Coherence Tomography (OCT), and skin lesion images for skin cancer classification.

Kermany et al. have used the chest X-ray images and OCT dataset to diagnose and treat diseases based on medical images by using deep learning [KGC+18].

Concerning the skin lesion images, we can take as an example the International Skin Imaging Collaboration (ISIC) that organises challenges [ISI]. These challenges deal with problems in lesion segmentation and detection of clinical diagnostic patterns and lesion classification. The goal of these challenges is often to build a classification model with lesion images. Since 2016, hundreds of participants have tried these challenges, making them the bigger standardised and comparative study in lesion classification and segmentation.

The various applications of DL models bring new vulnerabilities and a new way for attackers to get sensitive data and information. Attacks on DL can take multiple shapes and may have different goals. It can be some surveillance, some intentional tampering, or for illegal purposes as well. The attacks we have focused on are evasion or poisoning attacks.

2.2 Adversarial Attack

DNNs present several security concerns, and adversarial attacks represent a large part of them. Firstly, we will have a look at the evasion attacks, the figure.2.1 describe these attacks.

The goal for an attacker is to create an adversarial example that will evade the classifier. If we take an intrusion detection system, "evade" means to have our connection classified as "normal" instead of "intrusive". In the health domain, we could describe an evasion as classifying a benign lesion as a cancerous one and vice versa. We can perform such attacks by adding noise or modify the features of an image. It will impact the way the computer will "see" it. There are two types of image features, first are the global feature (e.g. colours and textures), and aims to describe an image as a whole. The second type is local features (e.g. points, edges, corners, or objects) and aims to detect key points or interest regions in an image and describe them [HAA16].

It is possible to attack with the well-known Fast Gradient Sign Method (FGSM) attack. The FGSM attack is one of the most known and a successful attack on DNNs. This attack uses the gradient of the neural network to compute an adversarial example that will evade the model [Zuo18]. FGSM perturbs the gradient direction of each feature by the gradient itself. This is done to maximise the loss, which is a value that evaluates the poorness of a specific classification in a DNN. This attack, as mentioned previously can lead to misclassification as illustrated in the figure.2.2 below.

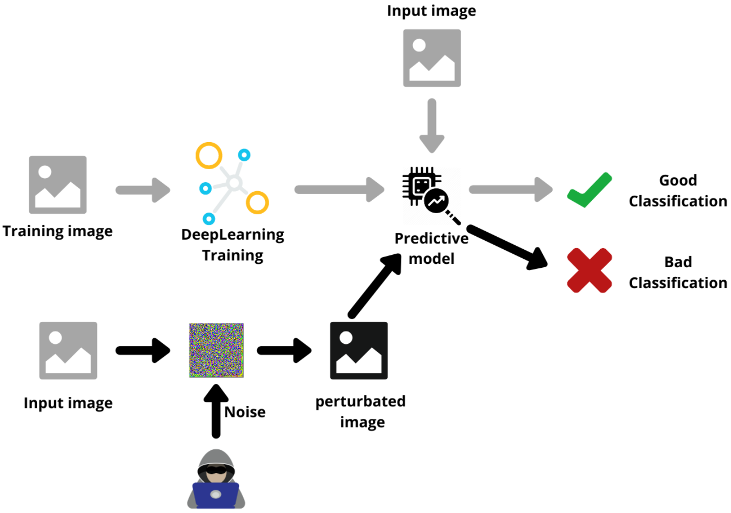


Figure 2.1: Adversarial attack phase

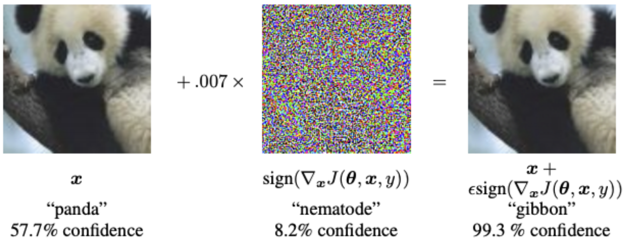

+ .007 ×
=
$\boldsymbol{x}$
"panda"
57.7% confidence
$\text{sign}(\nabla_{\boldsymbol{x}} J(\boldsymbol{\theta}, \boldsymbol{x}, y))$
"nematode"
8.2% confidence
$\boldsymbol{x} + \epsilon\text{sign}(\nabla_{\boldsymbol{x}} J(\boldsymbol{\theta}, \boldsymbol{x}, y))$
"gibbon"
99.3 % confidence

Figure 2.2: Misclassification of a panda performed with the FGSM Attack

In a 2017 paper, Carlini & Wagner presented a new method to perform adversarial attack [CW17]. The attack method is close to the FGSM previously introduced, as the idea is to find the minimum noise to add to an image that will change its classification. The specific feature in their method is that they define the distance between the input image and the computed attacked image. The difference with FGSM is that they want to minimise this distance. Their powerful attack defeated a specific mitigation method: the defensive distillation, and like they said: "Our attacks more generally can be used to evaluate the efficacy of potential defences".

Continuing with the evasion attacks, researchers have found a new kind of attack recently. This adversarial attack is an universal one. It means that applying a specific noise to any image will lead to a misclassification with a high percentage of chance. These are called Universal Adversarial Perturbation (UAP) [MFFF17]. Moreover, UAP is doubly universal first in terms of images, this attack works with almost every sample. Secondly, researchers have proven that UAPs can be used to attack any ML & DL model, which makes it universal in terms of models. Some papers prove that health-oriented DNNs are vulnerable to these adversarial perturbations [HMT20]. They have also presented a working mitigation method which is adversarial retraining. However, their process to retrain the model is quite complex and requires time (e.g. they use 10 DL models to compute UAPs and use these UAPs to perform the adversarial retraining mitigation method).

Moving on to poisoning attack; in those attacks the idea for an attacker is to add skewed data in the targeted dataset to compromise the machine learning algorithm (see figure.2.3).

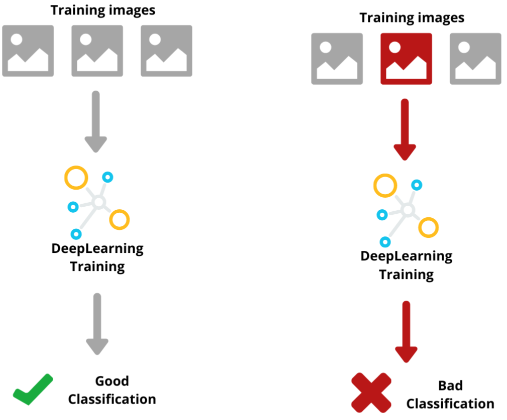


Figure 2.3: Poisoning attack explanation

These skewed data may be only images with the modified labels or some adversarial examples injected directly into the dataset. Hence, the algorithm will learn false features and will be less accurate than it would usually be.

A method to perform a systematic poisoning attack for machine learning in healthcare has been presented by Mozaffari-Kermani et al. [MKSKRJ15]. In this method, they add adversarial samples to the dataset. The adversarial images are generated by looking for attributes that match the statistics of the attacked class, but we set the label to the attacking class. This kind of attack is a targeted attack because we target a specific class while others remain untouched.

As previously mentioned, the growth of adversarial attacks targeting ML, and more precisely the healthcare, forces us to improve the security of these systems that may be under attackers' pressure.

2.3 Mitigation & Countermeasures

To mitigate previously introduced attacks, several methods exist, and a survey on countermeasures in machine learning has been conducted [QQBAF20]. We can split these countermeasures into three classes: modifying the model, modifying the dataset, and using an additional model. This categorisation is not specific to Healthcare it is slightly the same in other fields like the connected and automated vehicles for instance [QUQAF20].

Some of these countermeasures rely on pre-processing the input before going through the ML model while others involve adding robustness in the model itself.

– **Modifying the model:** The idea here is to add robustness into the model itself it can be by modifying parameters or features. A method such as defensive distillation means transferring the knowledge from a model to another one [HVD15]. The authors used the predicted labels of the first model as the labels of the input sample to their original model. This way, they increase the robustness of the DL model. Another method presented in the literature is gradient regularisation [RDV17]. It means: to obligate the gradient of our loss function to behave in a certain way, by adding structural constraints for example. These constraints may be various; it could for instance be telling the model which areas of the input are essential and which are not. [RHDV17].

– **Modifying the data:** These types of countermeasures modify either the data or its features. Catak et al. have presented a commonly used method such as adversarial retraining in their paper [CYY20]. As you can see in the figure.2.4, the idea is close to the poisoning attack previously described.

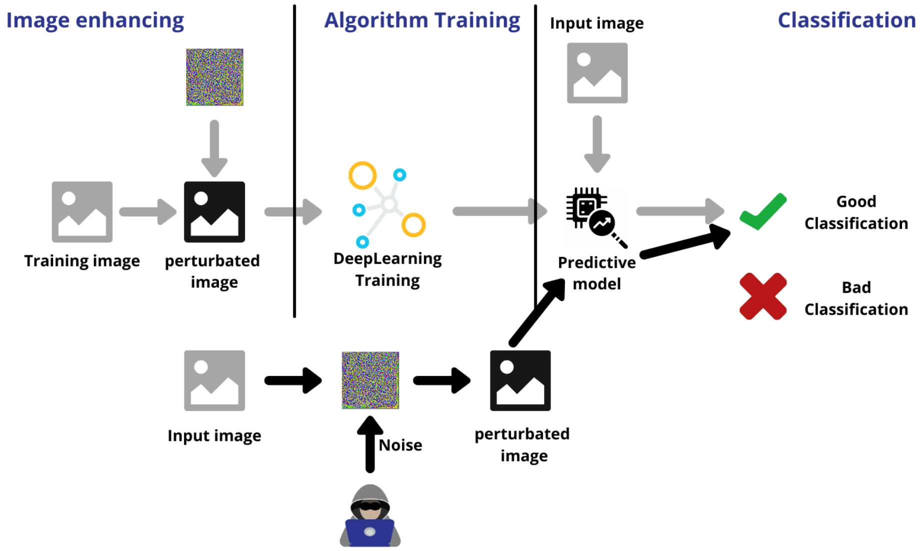


Figure 2.4: Adversarial training

Adversarial retraining means adding adversarial samples into the dataset but with the right label before retraining the model. By retraining, it means taking the first operational model and performing more epochs but with a new dataset for instance. Thus the model will learn the modified features and will be more robust once confronted with real adversarial examples. In the paper [CYY20], they tested this method against three types of attacks and their results are good. However, even if the researchers have performed this mitigation method on health-oriented data, our dataset is slightly different, so it is interesting to see how this method will behave in our case.

As well as the adversarial retraining, another way to modify the data is to use "pre-processing" tools. It means that when an adversary tries to misclassify an input - by adding some noise as an example - their tool can modify the poisoned input and the model will correctly classify it. In the first paper, the method proposed is to synthesise an image from scratch. This new picture would almost "be identical in appearance and semantics to the original image" [LZL+20]. While iterating on the picture, this method adds a controlled noise to makes it clearer, and to finally have a noise that would not lead to misclassification. The next figure.2.5 shows how iterating a lot of time allows to change the noise from (a) to (b). It is clear here that the noise present on (b) will not lead to misclassification, as it overlaps perfectly the original picture.

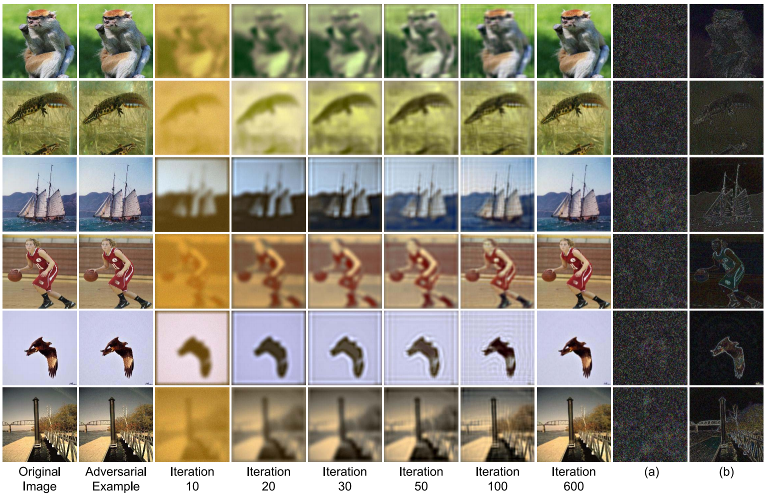


Figure 2.5: Online Alternate Generator Against Adversarial Attacks [LZL+20]

On another hand, the pre-processing module presented by Zhaoxia Yin et al. [YWWJT20], aims to destroy the adversarial noise that could have been added to the picture. It works in two-step, called "low-level transformation", a WebP compression and a flip. The compression is "specially designed to reduce the image details that are difficult to be perceived by human beings" [YWWJT20]. This means that possible adversarial noise, that is not visible by human eyes, would be reduced in that first step. The flip is used to "destroy the specific structure of adversarial perturbations" [YWWJT20], as flipping an image does not change the pixels themselves, but only their spatial position.

– **Additional model:** In these methods, to add robustness, additional auxiliary ML/DL models are integrated into the main model. We can mention often used method such as the use of adversarial detection. It is done by training an additional binary classifier to distinguish between the original and adversarial samples and can be regarded as a detector model [LIF17, GKPB20]. Another method belonging to this type of countermeasures is using generative ML models, this method was firstly described by Goodfellow et al. [GSS15] and the power of a Generative Adversarial Network (GAN) was described as well by Goodfellow et al. in another paper [GPAM+14]. The use of GAN has also been described by Santhanam et al. in their paper [SG18]. Their goal is to identify whether an input is adversarial or not and cleans it from the potential adversarial noise.

Chapter 3 Methodology

In this chapter, we detail the methodology to answer the research question described in the section.1.4 "Research Questions". The literature review is previously done and described in the chapter.2 "Related Work" allowed us to have a general overview of the problem and to build a detailed framework for our experiments. Before going into too many details about our experiments, we will look at the models we have used, the datasets and the different tools used in the experiments.

Before starting the experiments, we have built a framework to look at the work we have had to do. Here is the first and original framework :

- Implementing an image classification model (DNN)
- Adding noise to images to perform misclassification
- Trying to mitigate the attacks

Although this framework is basic, it highlights the main milestones of the experiment part, and we will detail each item in the following methodology section.

3.1 Models & Datasets

3.1.1 Models

After having built the framework and defined the objectives and the expectations of these experiments, we needed to select the models, datasets, adversarial attacks and countermeasures we would like to use. Pair work allowed us to split the work and consequently, perform more experiments. In terms of models, we came across two well-known models throughout our literature review, used with almost the same dataset [LL18] or used in a CoVid detection system based on X-ray images [CPM20]. Thus, we have decided to experiment on these two models: the Residual Net 50 (e.g. **ResNet 50**) and a Google Net the **Inception V3**. The purpose of working on two models is to ensure that results obtained on one model are not an only coincidence or due to luck we could say. Both model architectures are based on CNN [KSH12, AMAZ17] and in the figures below, you can see the architectures of both ResNet50 and Inception-V3.

The inception V3 architecture has been explained by Szegedy et al. [SVI+15, SLJ+14] and the two figures figure.3.1 and figure.3.2 shows the Inception V3 architecture and its specific feature, the Inception module. Inception modules are incorporated into CNN. The most simplified version of an inception module works by performing a convolution on input with not only one, but three different sizes of filters (1x1, 3x3, 5x5). It also performs a max pooling, as shown in the figure.3.2. Then, the resulting outputs are concatenated and sent to the next layer. This reduces the computation times of the whole network.

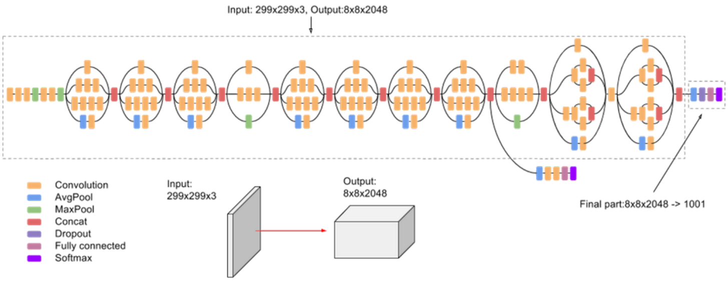


Figure 3.1: Inception V3 architecture

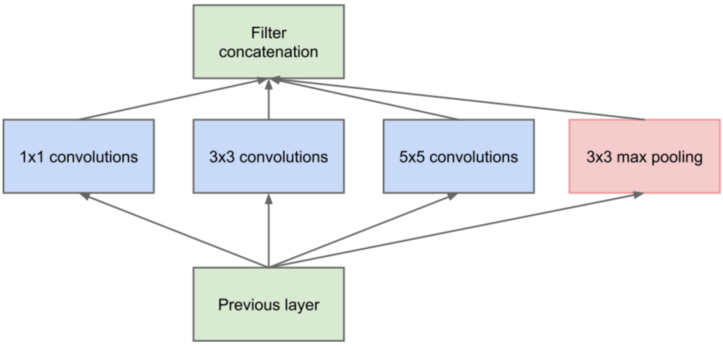


Figure 3.2: Inception module [SLJ+14]

The ResNet model, known as Residual neural networks, was developed by He et al. [HZRS16]. Its architecture is mainly based on residual blocks. According to Sabyasachi Sahoo, "in a traditional neural network, each layer feeds into the next layer. In a network with residual blocks, each layer feeds into the next layer and directly into the layers about 2–3 hops away." [Sah18]. The following figure shows how a residual block looks like.

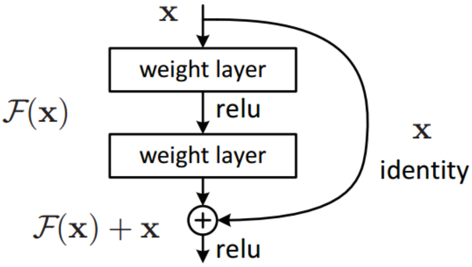
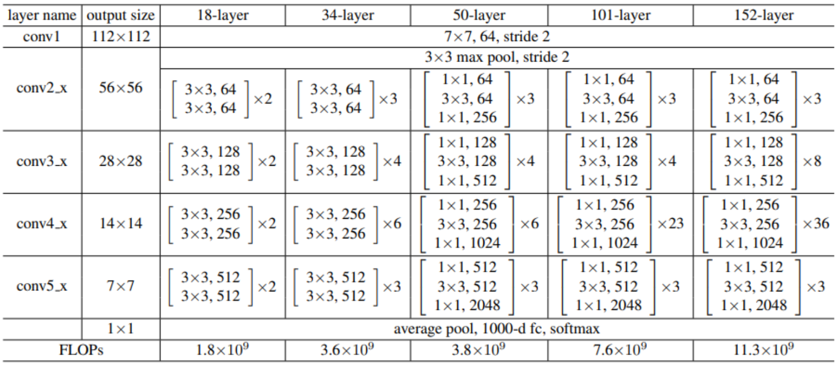

x
weight layer
$\mathcal{F}(\mathbf{x})$
relu
weight layer
x
identity
$\mathcal{F}(\mathbf{x}) + \mathbf{x}$
relu

Figure 3.3: Residual block illustration

Figure 3.4 illustrates the 50 layers of ResNet50. It shows the different layers that compose each kernel, and we can easily count on the fifth column: 1 + 3x3 + 4x3 + 6x3 + 3x3 + 1 = 50.



layer name	output size	18-layer	34-layer	50-layer	101-layer	152-layer
conv1	112×112	7×7, 64, stride 2				
conv2_x	56×56	3×3 max pool, stride 2				
		[3×3, 64 3×3, 64] ×2	[3×3, 64 3×3, 64] ×3	[1×1, 64 3×3, 64 1×1, 256] ×3	[1×1, 64 3×3, 64 1×1, 256] ×3	[1×1, 64 3×3, 64 1×1, 256] ×3
conv3_x	28×28	[3×3, 128 3×3, 128] ×2	[3×3, 128 3×3, 128] ×4	[1×1, 128 3×3, 128 1×1, 512] ×4	[1×1, 128 3×3, 128 1×1, 512] ×4	[1×1, 128 3×3, 128 1×1, 512] ×8
conv4_x	14×14	[3×3, 256 3×3, 256] ×2	[3×3, 256 3×3, 256] ×6	[1×1, 256 3×3, 256 1×1, 1024] ×6	[1×1, 256 3×3, 256 1×1, 1024] ×23	[1×1, 256 3×3, 256 1×1, 1024] ×36
conv5_x	7×7	[3×3, 512 3×3, 512] ×2	[3×3, 512 3×3, 512] ×3	[1×1, 512 3×3, 512 1×1, 2048] ×3	[1×1, 512 3×3, 512 1×1, 2048] ×3	[1×1, 512 3×3, 512 1×1, 2048] ×3
	1×1	average pool, 1000-d fc, softmax				
FLOPs		1.8×10^9	3.6×10^9	3.8×10^9	7.6×10^9	11.3×10^9

Figure 3.4: ResNet50 Detailed Architecture [KH15]

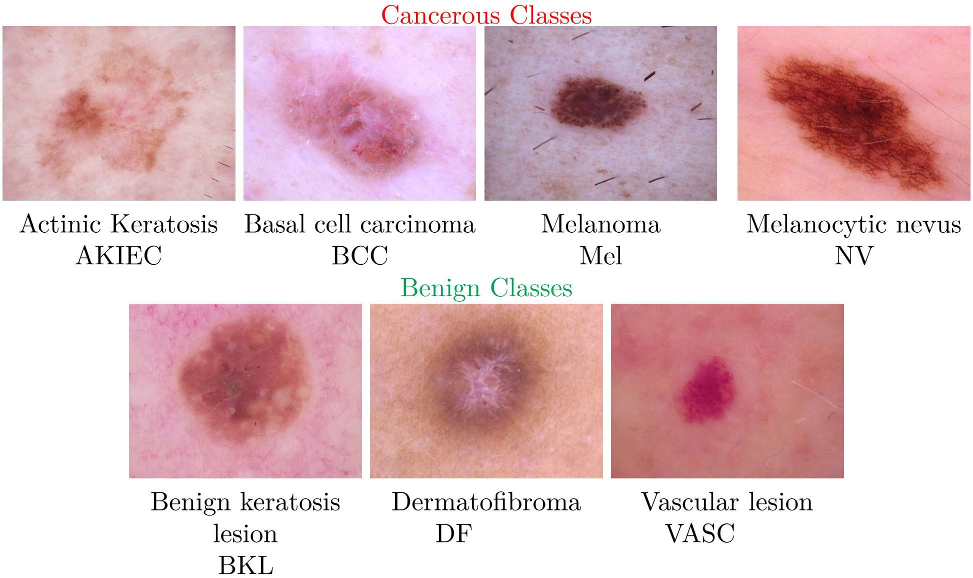
3.1.2 Dataset

In terms of datasets, as mentioned in the chapter.2 "Related Work", we have decided to use the skin lesion images dataset from the ISIC 2018 challenge about skin lesion analysis towards melanoma detection [LL18, ISI].

We have decided to take this dataset and not the chest X-ray images, as our supervisor is mainly specialised in this field and more precisely in melanoma classification. Also, as we had a lot of experiments, we could not perform all of them on several datasets and had to pick one. The ISIC 2018 is based on another dataset the HAM10000 [Tsc18]. Throughout our literature review, several papers mentioned the ISIC 2018 in their experiments. This, as well as the low weights of the overall dataset, made us chose this dataset over the ISIC 2019 or 2020 dataset.

The ISIC challenge, as mentioned in the chapter.2 "Related Work", gather multiple datasets in the health domain and more particularly in the skin lesion. This dataset from 2018 contains 10015 images and a metadata file with the patient data but our master's thesis is not to focus on the image classifier so we are not using these data.

The 10015 images are distributed into 7 classes which are detailed on the next page in the figure.3.5, this figure shows the input of our models. Moreover, the figure.3.6 and the figure.3.7 show the classes distribution in the dataset as well as the cancerous and healthy distribution.



Cancerous Classes

Actinic Keratosis
AKIEC

Basal cell carcinoma
BCC

Melanoma
Mel

Melanocytic nevus
NV

Benign Classes

Benign keratosis lesion
BKL

Dermatofibroma
DF

Vascular lesion
VASC

Figure 3.5: Classes samples in the ISIC 2018 dataset

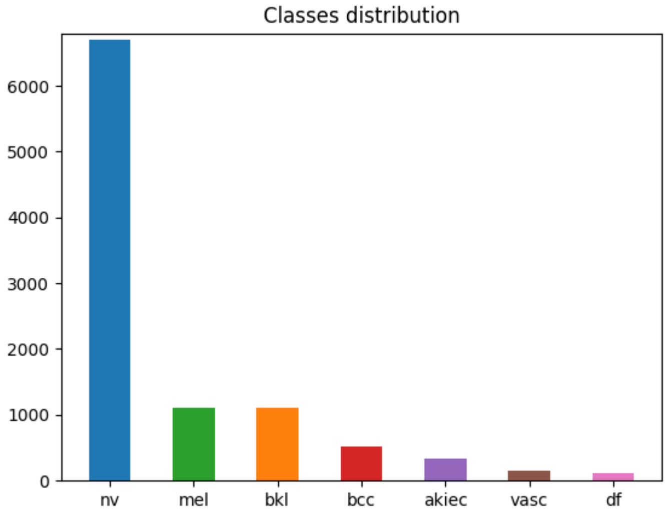


Figure 3.6: Classes distribution in the ISIC 2018 Dataset

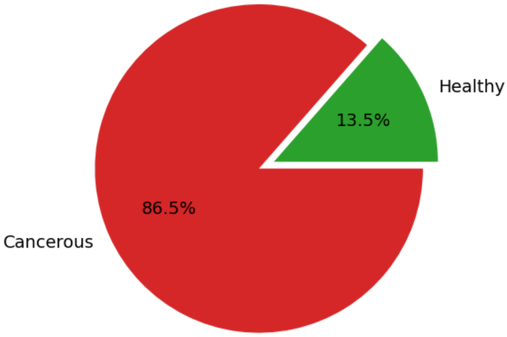


Figure 3.7: Cancerous and Healthy distribution in the dataset

As you can see from the previous figures, most of the dataset corresponds to a single class and this class correspond to a cancerous type of skin lesion. Thus, there is a problem of data imbalance in our dataset. We can solve this problem by introducing the class weight in the algorithm. It reduces in a certain way the impact of the most significant class and increases the impact of the others.

For the following experiment, we have decided to split the dataset as you can see in the figure.3.8. We took 70% for the training, 17.5% for the validation and 12.5% for the testing. This way it ensures that we test the model on images that have never been through the model.

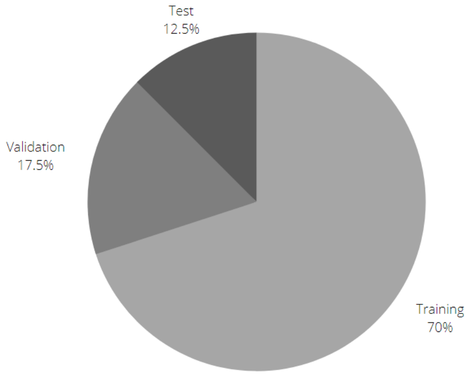


Figure 3.8: Dataset split

The use of these different datasets and the steps in machine learning have been well defined by Ripley in his book [Rip96].

"The training dataset is used to fit the model (The model will learn from the data). We use the validation dataset to provide an unbiased evaluation of a model fit on the training dataset while tuning model hyper-parameters. The evaluation becomes more biased as the training progress. Finally, the algorithms use the test dataset to provide an unbiased evaluation of a final model fit on the training dataset."

3.2 Adversarial Attacks & Countermeasures

In our project, we have decided to focus more on data and data manipulation. Thus, among all the adversarial attacks presented in the chapter.2 "Related Work", we have decided to take the FGSM, for its ease of implementation and its impact on the DL classifier. As mentioned in our research questions, we have also decided to implement UAPs because this adversarial attack is recent and powerful as it often drops classification results near zero. As seen in the chapter.2 "Related Work", they have developed a complex method to mitigate UAPs. Hence, it would be interesting to investigate and find an easier method to mitigate UAPs.

Concerning the countermeasures, we wanted to investigate and see if we could mitigate powerful attacks with a simpler method or by combining methods. We also wanted to focus our work on data and data manipulation, therefore we have decided to implement the adversarial retraining countermeasures and the use of LLT to remove the noise of our input image.

These two countermeasures have the benefit to be easy to implement and both modify the data. Hence, we could improve our experimental results by modifying the model itself or by using other methods presented in the chapter.2 "Related Work".

3.3 Keras & Tensorflow

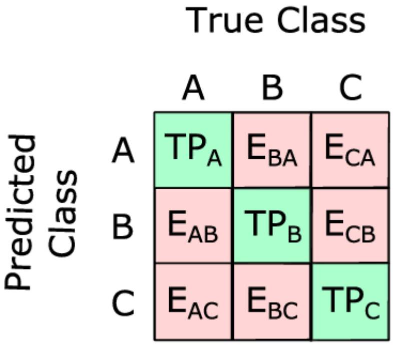
To perform experiments, we decided to use Python with Keras and Tensorflow libraries. Although we had never used these libraries, there are numerous guides and tutorials about it and a great community that shares their experiences. In order to gain in computation time we have used the TensorFlow-gpu as we were using a GPU cluster on a virtual machine provided by the university.

3.4 Performance Metrics

We have used several performance metrics to evaluate our different models during all the experiments and maintain a certain coherence while comparing our results. To evaluate our methods, we distinguish between the performances of the model itself and he risk that they represent for a patient in case of misclassification. For both kinds of evaluation, we have decided to use specific performance metrics. Every metrics we have used are based on the confusion matrix. To evaluate our methods, as mentioned, we have computed two different confusion matrices that will be detailed later.

3.4.1 Model performances

To evaluate the performances of our different models and methods, we firstly have used a multi-class confusion matrix. We can visualise the different values (e.g. True and False positive as well as negative) in the confusion matrix of figure.3.9.



	A	B	C
A	TP_A	E_{BA}	E_{CA}
B	E_{AB}	TP_B	E_{CB}
C	E_{AC}	E_{BC}	TP_C

Figure 3.9: Confusion matrix multi-class [Tha18]

Several metrics exist but we have chosen to use only a few of them. Bagli et al. have presented all metrics, their advantages and their uses have in their paper [GBV20].

Here are the metrics we have chosen to use:

- **Accuracy**

$$Accuracy = \frac{\text{Number of correct predictions}}{\text{Total number of predictions made}}$$

The accuracy gives an overview of the model performances. However, we can not evaluate the model performance by only using the accuracy. Even more in our case where we have an imbalanced dataset.

- **Recall** or **Sensitivity**

$$Recall_k = \frac{\text{TP}}{\text{TP}_k + \text{FN}_k}$$

The Recall measures the proportion of positive samples that were correctly classified for the specific class. It is slightly the same as the accuracy, excepted that each class has the same weight and importance.

This formula gives the recall for each class and from there we have computed the macro average recall :

$$MacroAverageRecall = \frac{\sum_{k=1}^{K} Recall_k}{K}$$

- **Specificity**

  Specificity often called the inverse recall represents the proportion of negative samples that were correctly classified.

$$Specificity_k = \frac{\mathrm{TN}_k}{\mathrm{FP}_k + \mathrm{TN}_k}$$

  We computed the macro average specificity

$$MacroAverageSpecificity = \frac{\sum_{k=1}^{K} Specificity_k}{K}$$

Recall and specificity highlights the performance class by class. Hence, if one of these metrics are low, it means that one or more classes have bad results.

3.4.2 Impact on Health

As mentioned before, the evaluation of a specific model or method in the health area is needed. Whether it is to understand the issue of such classification or to evaluate the risk after each classification, we have computed a second confusion matrix, as well as metrics based on the paper by Tohka et al. citeEvaluationModelHealth.

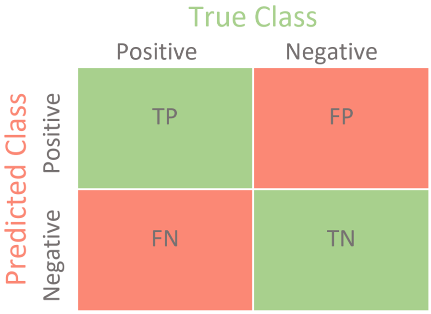


Figure 3.10: Confusion matrix simple case

In a health domain we can define these terms for a diagnostic as :

- True positive: Sick people **correctly** identified as sick
- False positive: Healthy people **incorrectly** identified as sick
- True negative: Healthy people **correctly** identified as healthy
- False negative: Sick people **incorrectly** identified as healthy

Here if we talk of the dangerous nature of such classification, the worst case in health domain is a false negative. If we have a false negative classification and there is no double-check or verification by the medical corps, it could lead to the development of the disease and could be deadly.

On the other hand, a false positive classification would lead to a deeper medical examination, which is in fact less dangerous but could be also stressful for the patient.

These values extracted from the confusion matrix are used to compute almost the same metrics like the one described before.

– **Accuracy**

$$Accuracy = \frac{\text{Number of correct predictions}}{\text{Total number of predictions made}}$$

The accuracy gives an overview of the model performance. However we can not trust the model only using the accuracy. Even more in our case where most of the images belong to one class.

– **Recall** or **Sensitivity**

$$Recall = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

In a medical environment it means that the model correctly detect ill patient that have a disease.

From the recall, we can get the false-negative rate which is: 1 - Recall.

– **Negative Predictive Value (NPV)**

If we consider the example of a medical test for diagnosing a disease, NPV tells how much we should 'believe' the classifier when it indicates that the person is healthy.

$$NegativePredictiveValue = \frac{\text{TN}}{\text{TN} + \text{FN}}$$

With the medically oriented confusion matrix, we have found it more important to have the NPV instead of the specificity, since we want to avoid false negatives.

Now that everything important related to the experiments and their performances have been presented, we can dive into the experiments' methodology.

3.5 Experiments

The figure.3.11. details the amount of experiments planned.

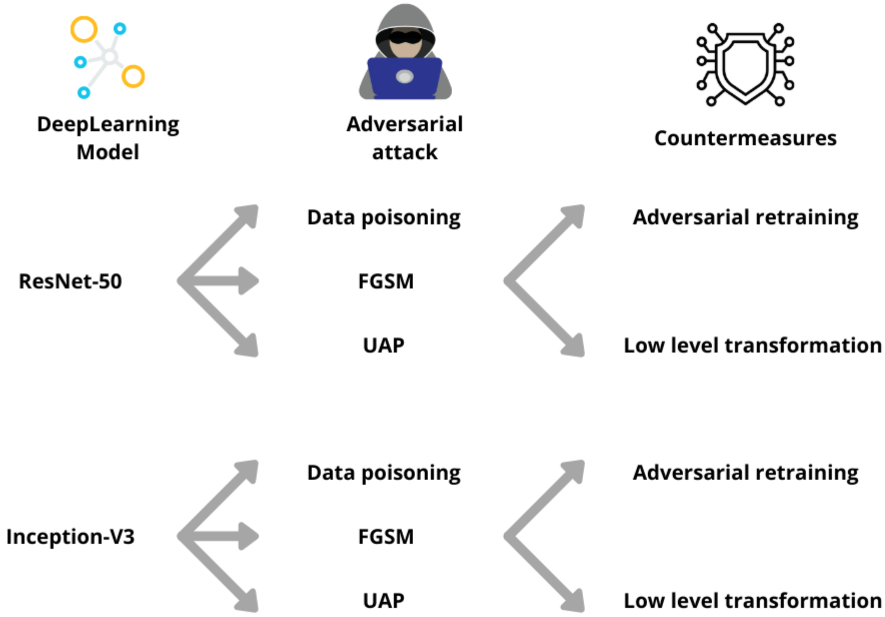


Figure 3.11: Experiments methodology

The first step was to implement an operational skin-lesion classifier, with the two models evoked earlier, the ResNet50 and the Inception V3. Once this implementation done, we moved on to the adversarial attacks part and set up 3 attacks for each model.

The idea of the first attack, the data poisoning, is to understand how we can dupe the classification model by adding skewed data to the dataset and the two others are evasion attacks. These attacks aim to lead the model to misclassify by adding perturbation to the image during the test process. We have generated these perturbations with FGSM and UAP, and we have described both methods in the chapter.2 Related Work.

We wanted to mitigate these attacks once they were effective. Thus, the last experiments are on the countermeasures we decided to implement. The two methods are adversarial retraining and the use of LLT as mentioned before. Like the attacks, both mitigation methods are described in the chapter.2 Related Work. As a final experiment, we wanted to combine both mitigation methods and see if it improves the results.

As we have decided to implement every attack on both models and the two countermeasures on both evasion attacks we ended up with a total of 14 experiments (+4 with a combination of both countermeasures). This amount of experiments allowed us to be more critical about the results we could have, and easily compare our models or adversarial attack methods or even mitigation methods between them.

Chapter 4
Experimental results

In this this chapter, we walk through the experiments we have performed, bringing details on classification, attacks and mitigation models used, as well as giving you an insight of the experimental results, with a brief first discussion about them.

4.1 Experiment 0: Operational skin lesion classifier

The idea of this first experiment was to become familiar with the different libraries presented, such as keras and tensorflow, as well as the different models' architecture (e.g. Inception V3 and ResNet-50) and, of course, the 2018 ISIC challenge dataset. In this experiment, we aimed to create two operational skin lesion classifiers. To achieve this goal, we have performed transfer learning on both models to reduce the computation time [ZQD+20].

To train our models we have used a batch size of 32 and we have trained for 50 epochs for both models.

In the beginning, our results were quite poor due to the lack of knowledge we had, but as things progressed, we have gained knowledge through the literature. Thus, the results improved. In the end, we achieved to have an operational skin lesion classifier on which we can perform our adversarial attacks and add the mitigation methods.

You can find below the different curves such as learning and validation loss as well as the accuracy during the training and validation for the two models we have trained.

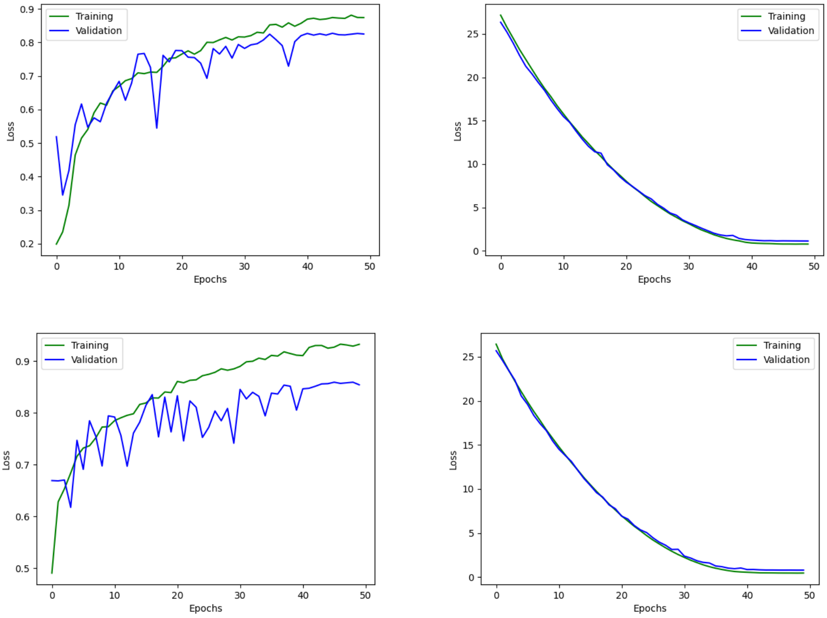


Figure 4.1: Accuracy (left) and loss (right) curves of the training for Inception V3 (top) and Resnet 50 (bottom)

These curves allowed us to monitor our learning performances and notice possible overfitting during the training of our models.

You can find on the next page the confusion matrix and the performance metrics for both models and the possible impact on the health of our classification.

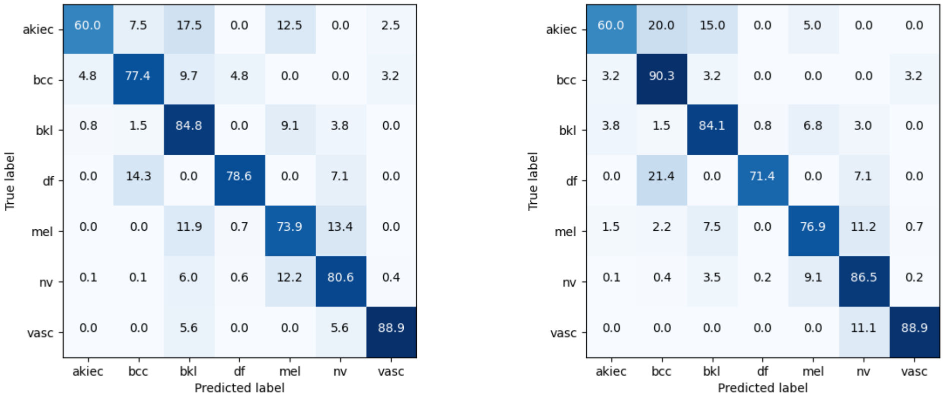


Figure 4.2: Confusion matrix classification model - Inception V3 (left) ResNet 50 (right)

Model	Accuracy	Sensitivity	Specificity
Inception V3	79.6%	77.8%	96.2%
ResNet 50	84.3%	79.7%	97.0%

Table 4.1: Skin lesion classifier performance

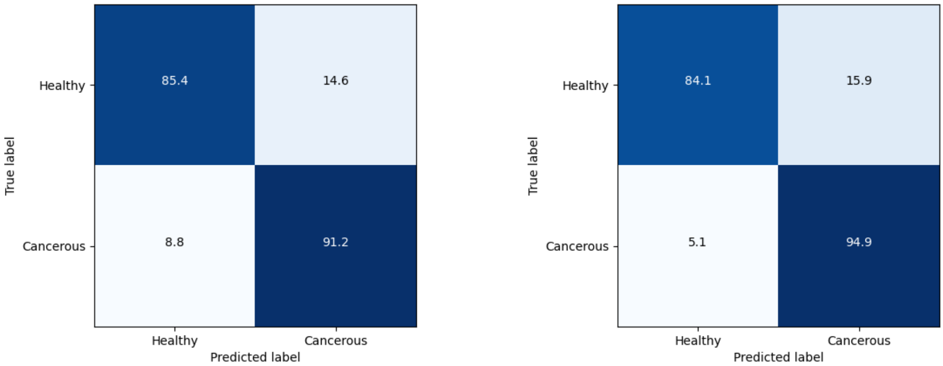


Figure 4.3: Health related confusion matrix - Inception V3 (left) ResNet 50 (right)

Model	Accuracy	Sensitivity	NPV	FNR
Inception V3	90.4%	91.2%	60.3%	8.8%
ResNet 50	93.4%	94.9%	72.3%	5.1%

Table 4.2: Skin lesion classifier health performances

We can see that most of the predictions made by the model are correct. Indeed, values in the diagonal represent the percentage of correctly classified pictures for each class. For example, InceptionV3 classified correctly 60% of the images labelled AKIEC. These models achieved an overall accuracy of 84%. Moreover, if we look at the health-related confusion matrix, the FNR is around 9 and 5% for our models, which is relatively good.

4.2 Experiment 1: Modify training dataset to perform misclassification

The first experiment concerning adversarial attacks involve modifying the training dataset previously described in the subsection 3.1.2 "Dataset". The attack's goal for us is to understand how we can dupe the model into bad classify images he would correctly classify before. The dataset modification will lead to a label modification assigned to some images. Because we are using supervised learning, this label modification will dupe the ML model into learning features of a specific class that are false. To modify the labels, we have just moved a certain percentage of a folder into another folder. During this experiment, we have chosen to modify the labels of images from the NV (Melanocytic nevus) class into the BKL (Benign keratosis) class. To see the real impact of such modifications we have decided to increase the amount of image moved from 5% of the NV class (approximately 335 images) to 10, 15, 20, 30, 45, and up to 60%.

You can find below figure.4.4 and figure.4.5 describing the overall idea of the dataset modification for the first four values (5 to 20% to understand the idea) and its impact on the classes distribution.

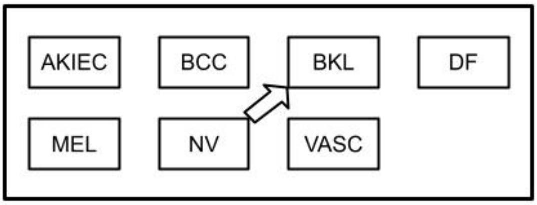


Figure 4.4: Modifying the training set by changing the labels

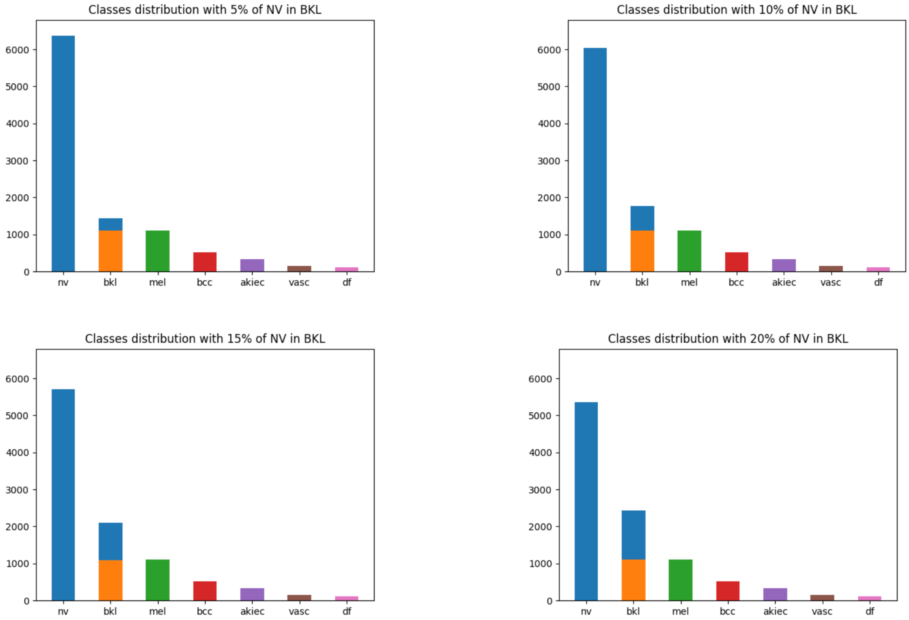


Figure 4.5: Modification of the dataset

As you can see on the last histograms, for the BKL class, only half of the images corresponds to the BKL class (when we have moved 20% of NV), the other half correspond to NV in reality. With this kind of modification, we want to observe some BKL images incorrectly predicted as NV, and it will highlight a decrease in the overall accuracy.

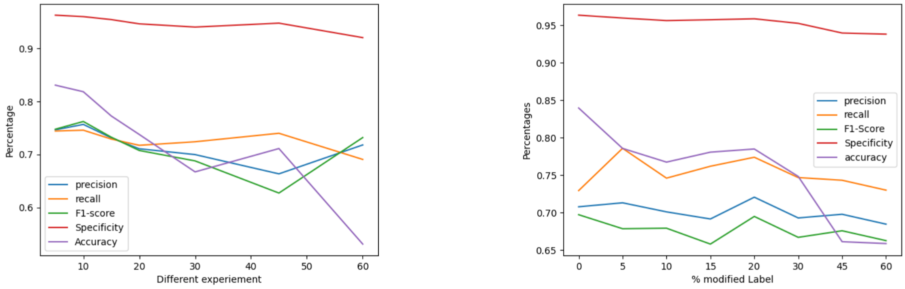


Figure 4.6: Metrics evolution over the experiment - Inception V3 (left) ResNet 50 (right)

In terms of results, you can find above in the figure.4.6 the performance curves that highlight the decrease of the accuracy and other performance metrics as the percentage of images that have their labels modified increases.

The same figure also shows that as the number of modified instances increases most of the metrics decreases. The specificity in these experiments will not be as impacted as other metrics because: "*Specificity represents the proportion of the negative samples that were correctly classified*" as mentioned in the section.3.4 "Performance Metrics". Hence, this attack goal is not to increase the number of false-positive, so the specificity does not drop as much as other metrics.

By manipulating the dataset, we have managed to create an operational poisoning attack. Our experiment was mainly a targeted attack, as we are using only two classes and the others remained untouched. The accuracy did not decrease for these classes. That is why the accuracy drops this much while the recall remains higher.

4.3 Experiment 2: Adding noise to perform misclassification

For the second experiment about adversarial attacks, we wanted to focus on evasion attacks. The idea is to add noise to our images to dupe the model into misclassification. To add the noise, we are going to use the two methods presented in the chapter.2 "Related Work": FGSM and UAP.

4.3.1 Experiment 2.1: The FGSM Attack

Firstly, as we have found more documentation and tools to perform the FGSM attack, we decided to begin our attacks experiment with this method.

As mentioned before in the chapter.2 "Related Work" this method creates a noise based on the gradient of the loss function used in the model. In addition, here is the equation corresponding to the FGSM.

$$adv_x = x + \epsilon * sign(\nabla_x J(\phi, x, y))$$

In this equation, we can see an ε. This value used to multiply the signed gradient. It ensures that the perturbations are small enough that the human eye cannot detect them but large enough to fool the DNN. In this paper, they have presented several adversarial attacks as well as several epsilon values [MNG+20]. In their experiment, they have used performed the FGSM attack with different value of epsilon. They have obtained the best accuracy with an epsilon value of 2/255.

The figure.4.3.1 below shows the perturbation crafted with FGSM and three example images before and after adding the perturbations.

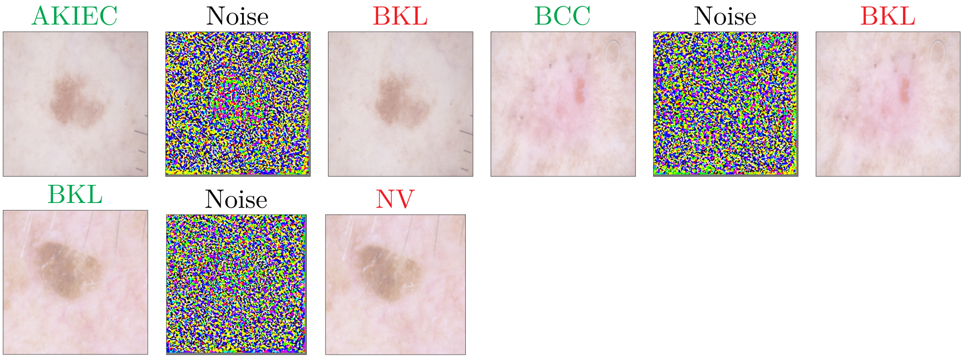


Figure 4.7: FGSM Noise and predictions

As we can see, it is impossible for us, humans, to detect these perturbations in any image. However, for the computer and the model, these images are different.

You can find below the different confusion matrix and tables with the performance metrics after we have attacked our models.

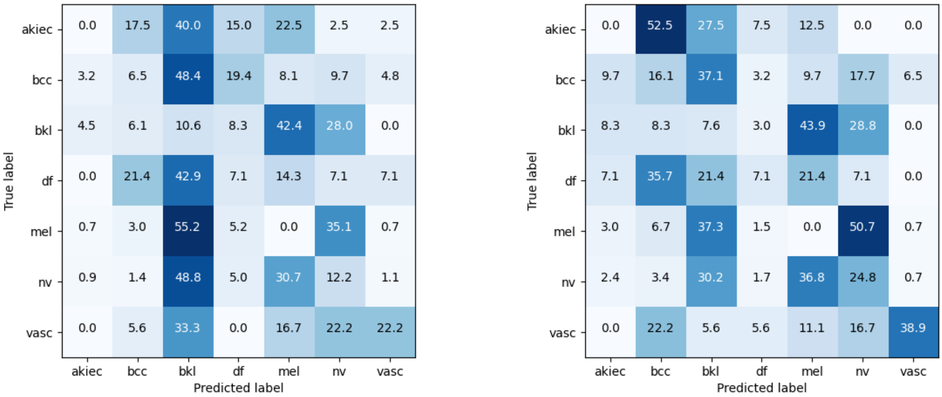


Figure 4.8: Confusion matrix after the FGSM attack Inception V3 (Left) & Resnet 50 (Right)

Model	Accuracy	Sensitivity	Specificity	Fooling rate
Inception V3	10.0%	8.4%	83.6%	90.0%
ResNet 50	18.9%	13.5%	84.4%	81.1%

Table 4.3: Experiment performance metrics after FGSM

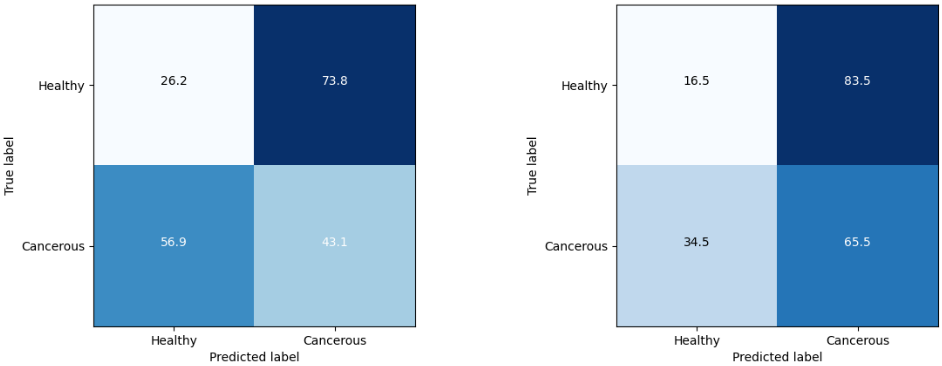


Figure 4.9: Health related confusion matrix after FGSM - Inception V3 (left) ResNet 50 (right)

Model	Accuracy	Sensitivity	NPV	FNR
Inception V3	40.8%	43.1%	6.7%	56.9%
ResNet 50	58.8%	65.5%	9.4%	34.5%

Table 4.4: Skin lesion classifier after FGSM health performances

We can see here the power of the FGSM attack; the accuracy of our model has dropped from around 80% to 15% if we look at the figure.4.8 unlike the first experiment most of the classification are out the diagonal. Moreover, concerning the health impact of such attacks, the FNR is high and the NPV low. It means that the model can not be trusted since even though the lesion is predicted benign, there is a very high probability that it is indeed cancerous. Trusting this model could be life-critical because of the possibility of undiagnosed patients.

4.3.2 Experiment 2.2: The UAP Attack

For the second experiment of attack, we have used UAPs. Like we mentioned in the chapter.2 "Related Work", UAP is a method to build a specific noise that will lead to misclassification once added to any image of the dataset.

How this method works in practice ?

UAP method iterate over the targeted dataset and, for every image in this dataset it applies a noise with a chosen method. It can be FGSM as we have used before, but it can be with any other method. If the attack does not lead to misclassification, the FGSM method - or other methods - generate a new noise based on the same image but with the previous noise added to it until the attack succeed. Once the noise added by the method leads to a successful attack, i.e. the prediction by the model is different from the correct label. The global noise is updated with the noise used to perform this attack, and so on for every image in the dataset. In the end, one "global" noise is computed and added to each image and will fool the model.

To generate these perturbations, we have used the method in the Adversarial Robustness Toolbox (ART) that you can find here :(*https://github.com/Trusted-AI/adversarial-robustness-toolbox/*)[NST+18]. You can find below the different results we have got with this method.

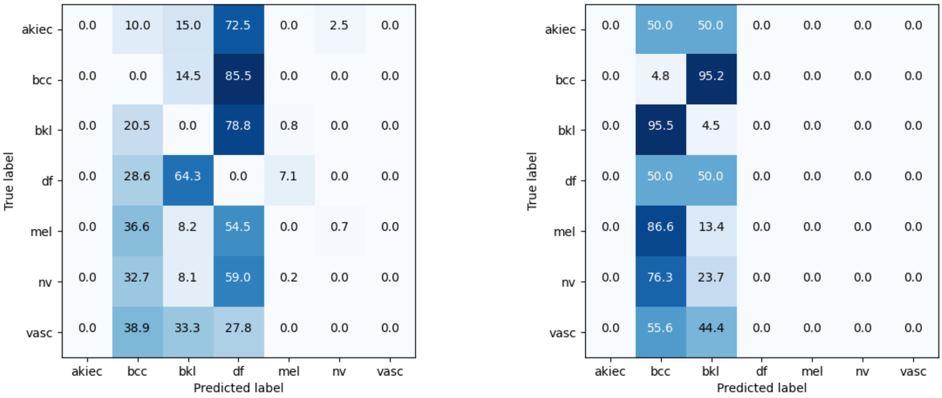


Figure 4.10: Confusion matrix after UAP, Inception V3 (Left) & Resnet 50 (Right)

Model	Accuracy	Recall	Specificity	Fooling rate
Inception V3	0.0%	0.0%	85.2%	100.0%
ResNet 50	0.7%	1.3%	84.8%	99.3%

Table 4.5: Experiment performance metrics after UAP

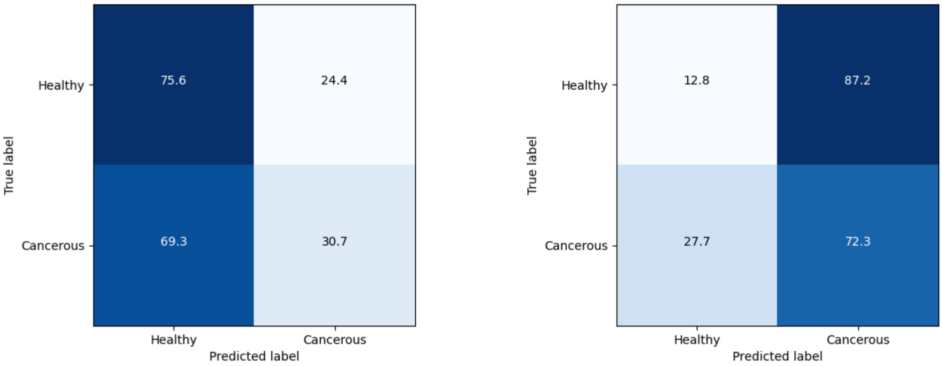


Figure 4.11: Health related confusion matrix after UAP - Inception V3 (left) ResNet 50 (right)

Model	Accuracy	Sensitivity	NPV	FNR
Inception V3	36.8%	30.7%	14.7%	69.3%
ResNet 50	64.4%	72.3%	6.8%	27.7%

Table 4.6: Skin lesion classifier after UAP health performances

As we can see on the figure.4.10 and figure.4.11 as well as the table.4.5 and table.4.6 the accuracy after UAP drops near 0 for both models which is even more powerful than the FGSM previously presented. We can also note that classification is mostly in one or two classes for each model, which means that the global noise computed with UAP is close to these classes.

If we quickly take a look at the health-related metrics, the classification for the Inception V3 is poor. Most predictions belong in the healthy classes, while for ResNet50, most predictions are in the cancerous classes. It is linked with the classification in only one or two classes for each model. The ResNet50 has classified most images as BCC, a cancerous lesion, while Inception V3 has classified most images as DF, a healthy lesion.

4.4 Experiment 3: Mitigate the attacks

We have previously shown how efficient attacks on DL models could be. This section will now focus on mitigating these attacks. We have planned to use some methods (e.g. Adversarial retraining and LLT), which researchers have already tested but on a different dataset with different models. Thus, regarding these methods, the goal is to see if they suit an ML model in the health domain. Concerning mitigating the UAP attack, that is one of our research questions, the goal is to see if we can mitigate them, and what would be the performances with a simpler method.

4.4.1 Experiment 3.1: Adversarial retraining

The first method we have planned to use to mitigate the attacks is adversarial retraining. This method - as mentioned in the chapter.2 "Related Work" - is close to the idea of poisoning attack, except that in this case, the labels from images stay the real labels to add robustness in our predictive model. To mitigate both the FGSM and UAP methods, we needed to retrain our models with both methods. The way we have retrained our model is simple. We have created an adversarial dataset by randomly taking 50% of the training and validation dataset then we have added perturbation - computed by FGSM or UAP - to every image. Then the previously trained model is retrained for five epochs on this adversarial dataset that contains the normal dataset and the adversarial samples added to it. We have repeated the adversarial dataset construction 5 times and the model just retrained is used as an input for the next retraining, for a total number of 25 epochs of retraining in the end.

Once we have retrained the model, we have tested its robustness against attacks. As mentioned in the *Experiment 2.1 : The FGSM Attack*, the method base the perturbation on the model itself (e.g. by using the loss function). To be more realistic, we suppose that the attacker has a copy of the classifier model (e.g. before the adversarial retraining) on which he can compute perturbations. Then these perturbations are added to images and tested on the retrained model as it would be in a real situation.

You can find the different confusion matrix and performance metrics below for each case after we have retrained our models.

Adversarial retraining FGSM

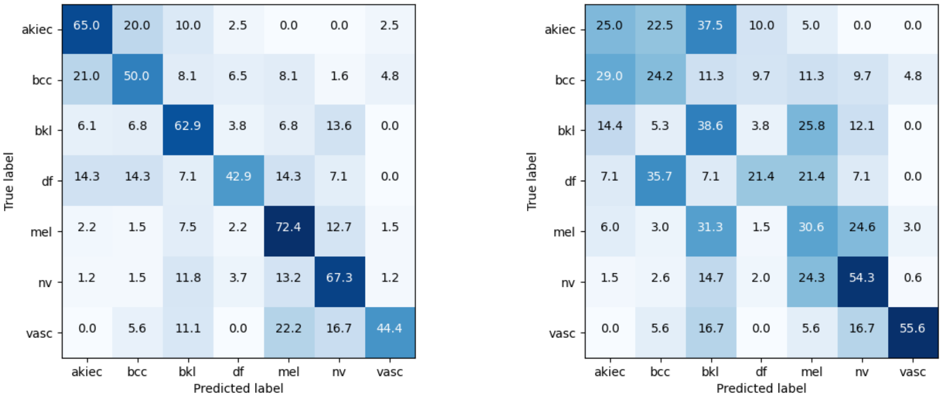


Figure 4.12: Confusion matrix FGSM after adversarial training Inception V3 (Left) & Resnet 50 (Right)

Model	Accuracy	Recall	Specificity	Fooling rate
Inception V3	65.8%	57.8%	93.8%	34.2%
ResNet 50	47.1%	35.7%	90.3%	53.0%

Table 4.7: Experiment performance metrics after adversarial training - FGSM attack

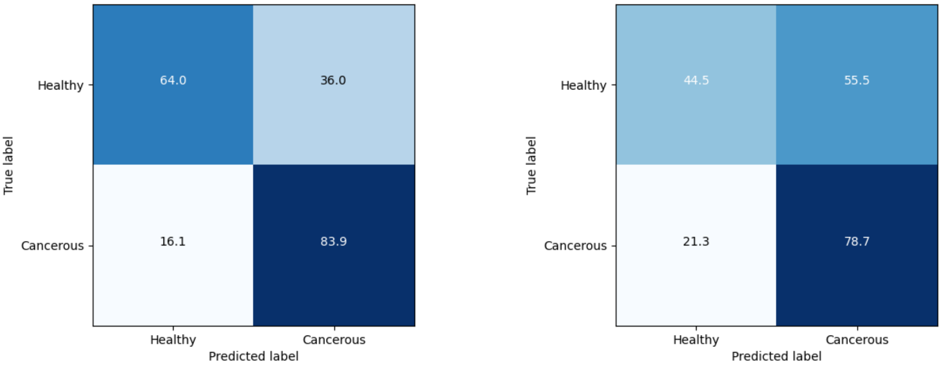


Figure 4.13: Health related confusion matrix after FGSM with adversarial training - Inception V3 (left) ResNet 50 (right)

Model	Accuracy	Sensitivity	NPV	FNR
Inception V3	81.2%	83.9%	38.5%	16.1%
ResNet 50	74.0%	78.7%	24.7%	21.3%

Table 4.8: Skin lesion classifier after FGSM with adversarial training health performances

Adversarial retraining UAP

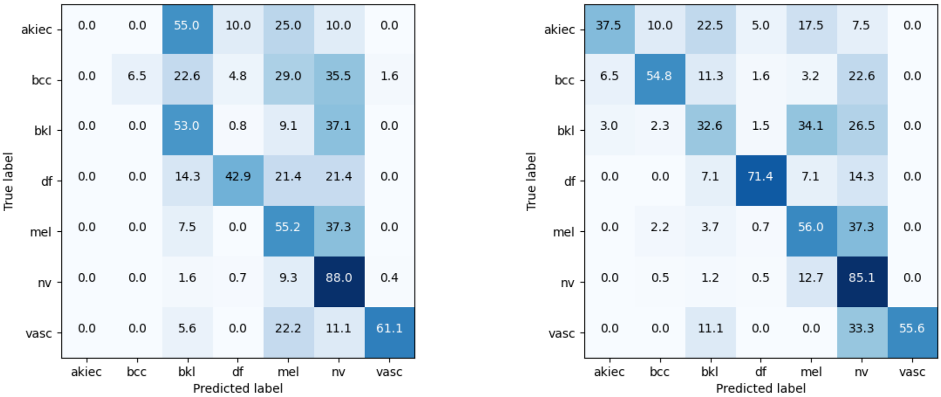


Figure 4.14: Confusion matrix UAP after adversarial training, Inception V3 (Left) & Resnet 50 (Right)

Model	Accuracy	Recall	Specificity	Fooling rate
Inception V3	72.5%	43.8%	92.7%	27.5%
ResNet 50	72.3%	56.1%	93.1%	27.6%

Table 4.9: Experiment performance metrics after adversarial training - UAP attack

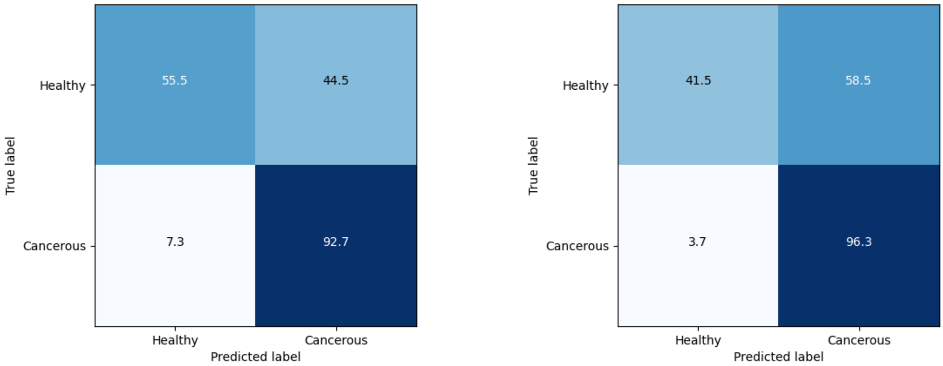


Figure 4.15: Health related confusion matrix after UAP with adversarial training - Inception V3 (left) ResNet 50 (right)

Model	Accuracy	Sensitivity	NPV	FNR
Inception V3	87.6%	92.7%	54.5%	7.3%
ResNet 50	88.8%	96.3%	63.5%	3.7%

Table 4.10: Skin lesion classifier after UAP with adversarial training health performances

As we can see with the different tables and figures above, the adversarial retraining improves the results we have had against both FGSM and UAP. The metrics that evaluate our models' performance increase even if it is still lower than in a normal situation. Moreover, if we look at the health-oriented matrix and metrics, this method better mitigates FGSM than UAP.

4.4.2 Experiment 3.2: Low level transformation

The second method we have planned to use to mitigate adversarial attacks is the use of LLT, this method - as mentioned in the chapter.2 "Related Work" - idea is to remove the perturbation in the image by using 2 low level operations. These operations are a webp compression and a flip to the image. This mitigation method works like a pre-processing tool, most adversarial images computed by using FGSM or UAP is compressed and flipped before being classified by the model.

You can find the different confusion matrix and performance metrics below for each case when every adversarial sample are pre-processed before being classified.

LLT against FGSM

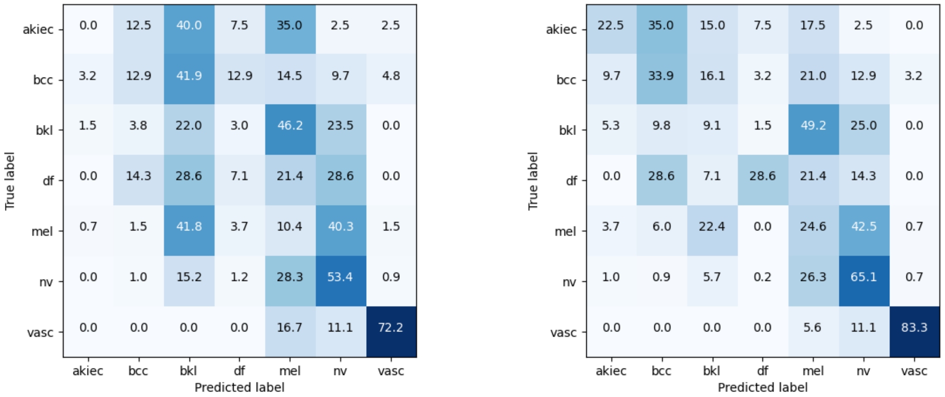


Figure 4.16: Confusion matrix FGSM after low level transformation, Inception V3 (Left) & Resnet 50 (Right)

Model	Accuracy	Recall	Specificity	Fooling rate
Inception V3	41.1%	25.4%	88.4%	58.9%
ResNet 50	51.3%	38.2%	90.0%	48.7%

Table 4.11: Experiment performance metrics after low level transformation - FGSM attack

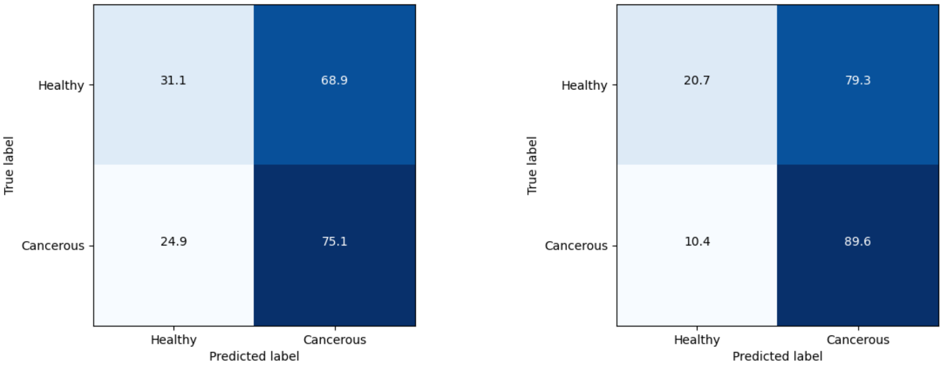


Figure 4.17: Health related confusion matrix after FGSM with low level transformation - Inception V3 (left) ResNet 50 (right)

Model	Accuracy	Sensitivity	NPV	FNR
Inception V3	69.1%	75.1%	16.5%	24.9%
ResNet 50	80.2%	89.6%	23.9%	10.4%

Table 4.12: Skin lesion classifier after FGSM with low level transformation health performances

LLT against UAP

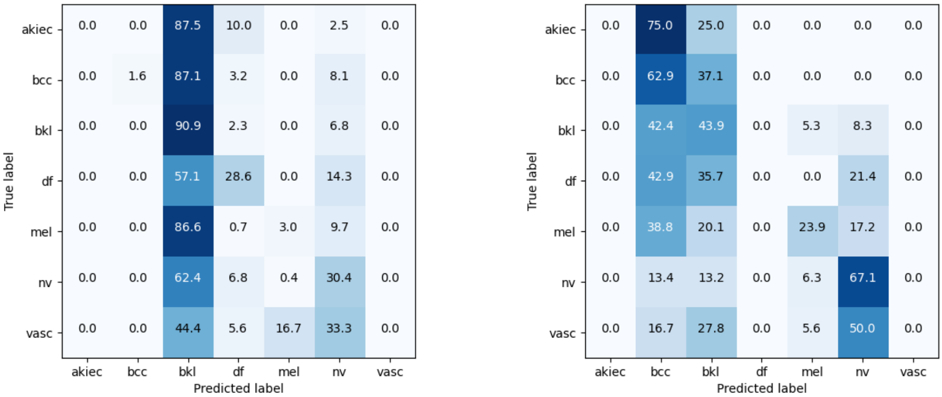


Figure 4.18: Confusion matrix UAP after low level transformation, Inception V3 (Left) & ResNet 50 (Right)

Model	Accuracy	Recall	Specificity	Fooling rate
Inception V3	31.0%	22.1%	88.2%	69.0%
ResNet 50	55.5%	28.3%	92.1%	44.5%

Table 4.13: Experiment performance metrics after low level transformation - UAP attack

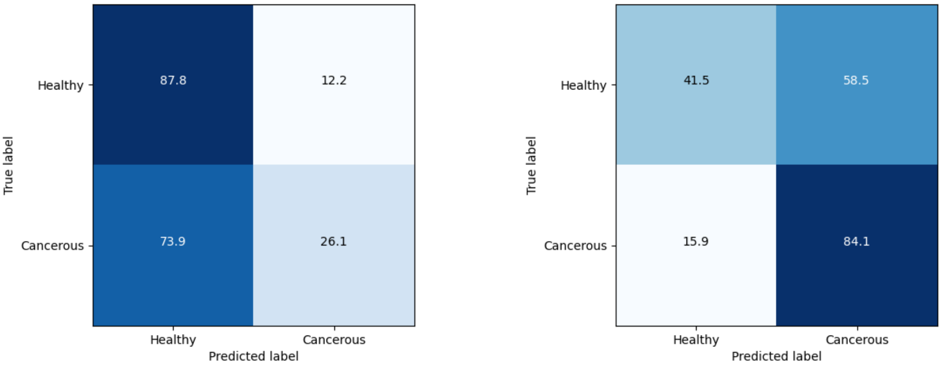


Figure 4.19: Health related confusion matrix after UAP with low level transformation - Inception V3 (left) ResNet 50 (right)

Model	Accuracy	Sensitivity	NPV	FNR
Inception V3	34.5%	26.1%	15.8%	73.9%
ResNet 50	78.3%	84.1%	29.1%	15.9%

Table 4.14: Skin lesion classifier after UAP with low level transformation health performances

As we can see with the different tables and figures above, the LLT slightly improves the results we have to add against both FGSM and UAP. However, we can see for the Inception V3 model the metrics do not increase as much as the ResNet 50. Due to the timing, we could not conduct more experiments to identify the source of these differences. Overall the LLT countermeasure improves the results for ResNet50 against both adversarial attacks but still not enough to be used alone.

4.4.3 Experiment 3.3: Combining both mitigation method

As mentioned in the Research Questions we aimed to combine both mitigation methods presented before. Instead of using the base model (e.g. Inception V3 or Resnet 50), we have taken both models retrained on the adversarial dataset as we have explained in the "Experiment 3.1: Adversarial retraining". Then we have added the LLT method to note the evolution of the metrics when we combine both mitigation methods.

You can find below all the figures and metrics representing our model performance and health-oriented metrics against FGSM and UAP attack once we have added the 2 mitigation methods (e.g. adversarial training and low-level transformation).

Adversarial retraining + LLT against FGSM

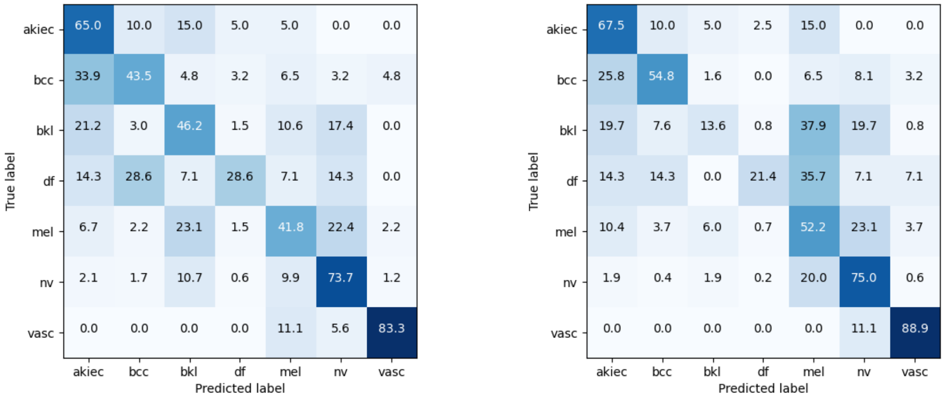


Figure 4.20: Confusion matrix FGSM after both mitigation method, Inception V3 (Left) & Resnet 50 (Right)

Model	Accuracy	Recall	Specificity	Fooling rate
Inception V3	64.9%	54.6%	93.2%	35.1%
ResNet 50	64.1%	53.3%	92.9%	35.9%

Table 4.15: Experiment performance metrics after both mitigation method - FGSM attack

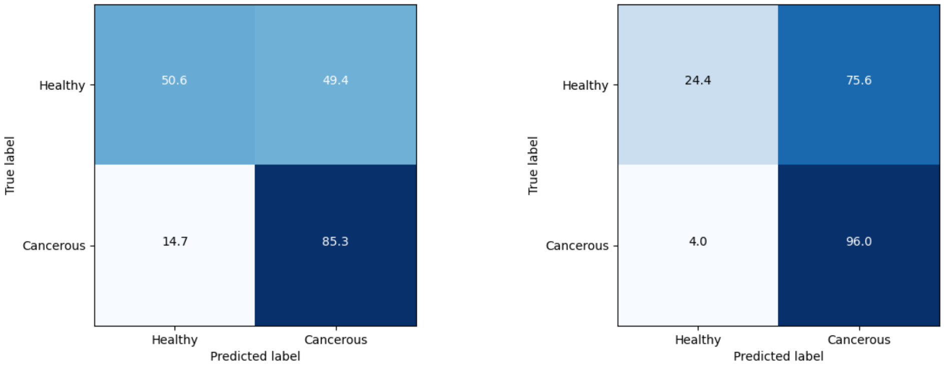


Figure 4.21: Health related confusion matrix after FGSM with both mitigation methods - Inception V3 (left) ResNet 50 (right)

Model	Accuracy	Sensitivity	NPV	FNR
Inception V3	80.6%	85.3%	35.2%	14.7%
ResNet 50	86.2%	95.9%	48.7%	4.0%

Table 4.16: Skin lesion classifier after FGSM with both mitigation method health performances

Adversarial retraining + LLT against UAP

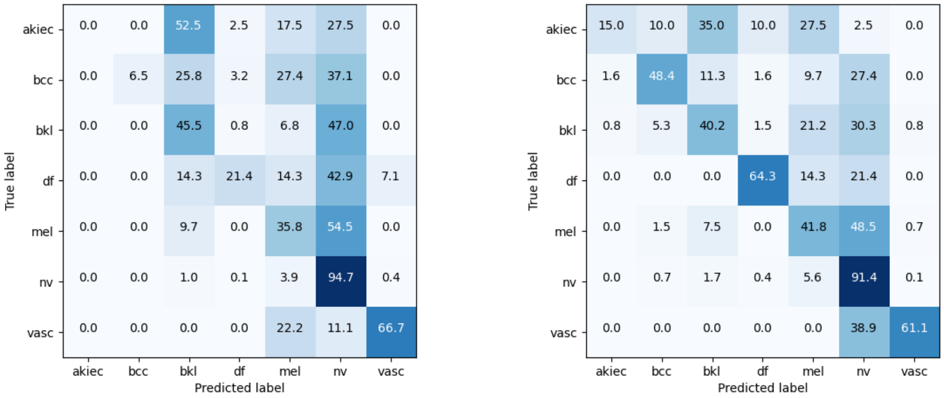


Figure 4.22: Confusion matrix UAP after both mitigation method, Inception V3 (Left) & Resnet 50 (Right)

Model	Accuracy	Recall	Specificity	Fooling rate
Inception V3	73.8%	38.6%	91.8%	26.2%
ResNet 50	74.8%	51.7%	93.0%	25.2%

Table 4.17: Experiment performance metrics after both mitigation method - UAP attack

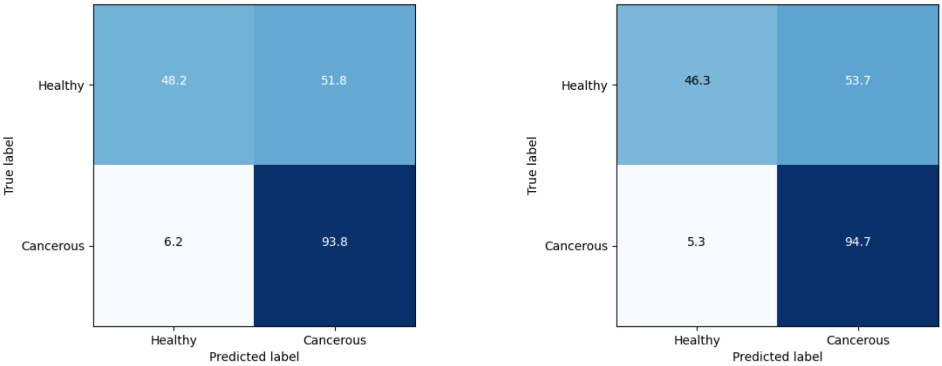


Figure 4.23: Health related confusion matrix after UAP with both mitigation methods - Inception V3 (left) ResNet 50 (right)

Model	Accuracy	Sensitivity	NPV	FNR
Inception V3	87.5%	93.8%	54.9%	6.2%
ResNet 50	88.1%	94.7%	58.0%	5.3%

Table 4.18: Skin lesion classifier after UAP with both mitigation method health performances

As we can see with the different tables and figures above, when we combined both mitigation methods that have given acceptable results before, we ended with even better results.

The results improve drastically against UAP and are close to the results before any attack. Moreover, the classifier has correctly predicted most cancerous lesion images as the figure.4.23 highlights.

However, if we take a look at the results against FGSM, table4.15, it shows that the model performances remain quite low. Even though the model has correctly classified most cancerous lesion images as cancerous, we could improve the overall results against this FGSM attack.

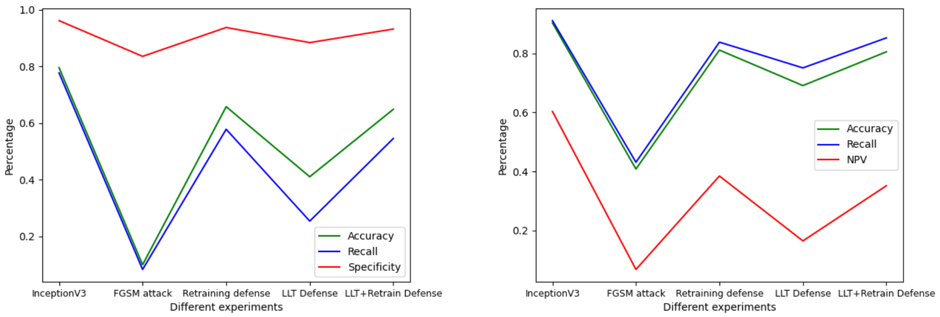
Chapter 5 Discussion

In this chapter we are going to perform a further analysis of the results for each experiment presented in the previous chapter, as well as giving the answers to the research questions presented in the beginning of this report.

5.1 Experimental results discussion

To discuss our results, we have computed curves of our results to have a more global view both in terms of model performances and impact on health as we mentioned in the section.3.4 "Performance Metrics".

Inception V3



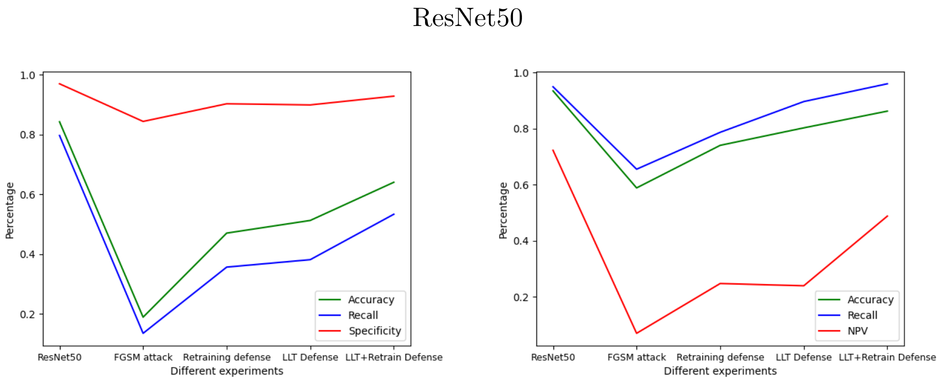
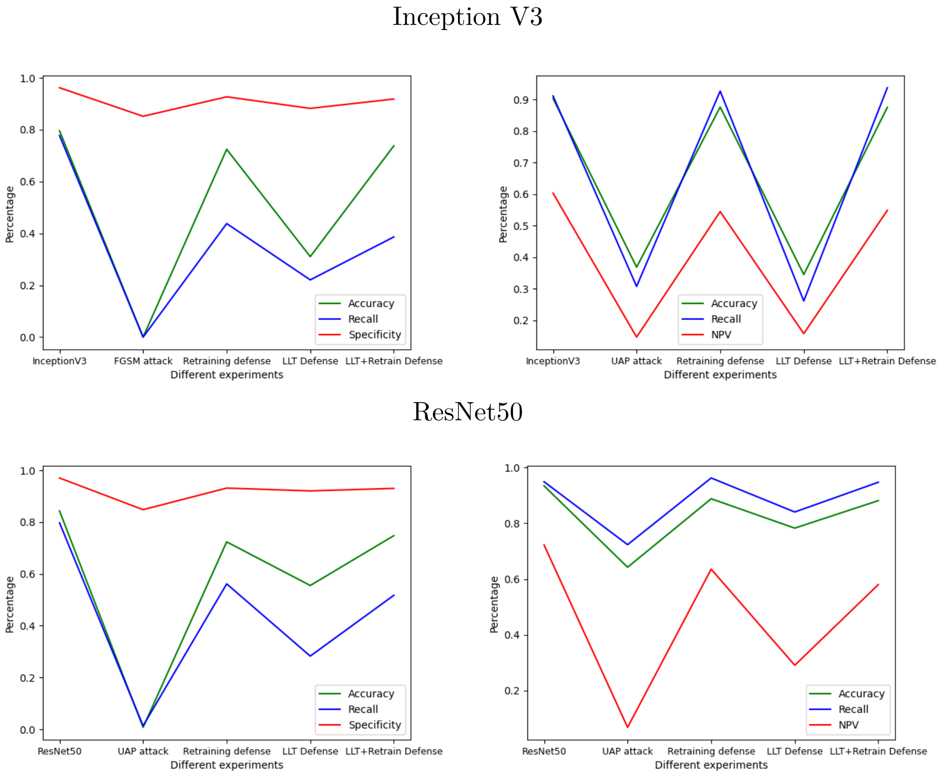

ResNet50

Figure 5.1: Recap curves against FGSM attack for model performances (left) and impact on health (right)


Inception V3

ResNet50

Figure 5.2: Recap curves against UAP attack for model performances (left) and impact on health (right)

The different figures above well recap the results we have got with our methods. We see that both attacks (e.g. FGSM and UAP) drop our metrics (e.g. the one used to evaluate our model performance or the impact on health) for both models. If we look at these metrics, there is a brief gap between recall and accuracy and between NPV and the other metrics. This gap is due to the imbalance in the dataset. As mentioned in the section.3.4 "Performance Metrics" the recall highlights when some classes have bad results. The accuracy is high because the most common class is well classified, while others may have terrible results. The figure.4.22 highlights this, where we have achieved 88% accuracy in the NV class (e.g. the most represented one in the dataset) while 0% and 6.5% for two other classes. The fig.5.2 shows this with the left curves that the recall is quite low while the accuracy remains around 80%. This problem is more obvious in the UAP curves because UAPs are more powerful than FGSM.

The same "issue" appears in the impact on health curves, where the NPV remains lower than other metrics. As mentioned in the section.3.4 "Performance Metrics", the NPV relates how much we should "trust" the classifier when it indicates that the person is healthy. Hence, if such metrics are low, people are likely to not trust our models, even if the FNR is low. Talking about the FNR and NPV, these values concern the health impact of the classification, where both metrics relate to the capability of the model to not misclassify cancerous lesions as healthy lesions. As shown in the recap curves, the recall remains high while the system is not under attack but the NPV is not that high. It means that the classifier has correctly classified most cancerous lesion images as cancerous but it has badly classified a small amount as healthy. However, with an imbalanced dataset, this small amount is large compared to the number of healthy images classified as healthy. Thus, the NPV can not be as high as the recall in the right curves.

If we ignore these metrics due to the data imbalance, we have achieved noteworthy results. The different recap curves well highlight the results' improvement after adding our mitigation methods. Whether it is for the model performance or the impact on health, the metrics nearly came back to their normal values after we have added mitigation methods (e.g. normal values means values before any attack on our models).

As mentioned during the "Experiment 2.1: The FGSM Attack" the results of our mitigation methods against FGSM are acceptable, yet not as good as the results against UAP as the figure.5.1 highlights. The health-oriented metrics came back to good values but the model performances remain lower than expected.

5.2 Research question discussion

In this research, we have attempted to answer the following research questions previously defined in the section.1.4 "Research Questions" :

RQ1: How would the classifiers - combination of DL model and ISIC2018 dataset - be impacted by adversarial attacks (e.g. FGSM, UAP)?

RQ2: Concerning adversarial attacks, more precisely for FGSM, is there a way to mitigate this attack on this dataset?

RQ3: Is there a way to achieve the same results against UAP (Presented by Hirano et. al. [HMT20]) by using a simplified method or by combining known methods (e.g. Adversarial training and Low-level Transformation)?

5.2.1 The answer to first research question

As we mentioned throughout this research paper, most attacks or countermeasures have already been presented in another field. The experiments were different in terms of models or dataset, for instance. Thus, the main goal was to transfer the methods (e.g. Adversarial attacks and countermeasures) we have presented in the Methodology chapter. We now can say that the ISIC 2018 dataset is vulnerable. As we can see in the different recap figures figure.5.1 and figure.5.2 presented in the section.5.1 "Experimental results discussion" before, the model performances under attack dropped to nearly zero.

5.2.2 The answer to second research question

Concerning the FGSM attack, especially on this dataset, we have managed to mitigate the attack. If we look at the model performances after each mitigation methods (e.g. Retraining and LLT), the metrics moderately increased. However, when we combine both methods, the metrics slightly increased but not as we expected. After all these experiments, we can say that we have achieved noteworthy results, even though there is still room for improvement for this question.

5.2.3 The answer to the third research question

As we have mentioned in the chapter.2 "Related Work", Hirano et al. presented a method to mitigate UAPs. However, their method is quite complex as it requires ten models trained to perform UAPs. Thus, we wanted to mitigate UAPs, by using a simplified method (e.g. simpler adversarial retraining or LLT). Once we have combined both countermeasures, whether it is the model performances or the health-oriented metrics, both accuracy of our models came back to appropriate values, as we can see in the figure.5.2. However, this high accuracy is not enough to evaluate our method. As we have mentioned in the last section, there is a gap between accuracy and recall which indicates that the model has correctly classified the most representative classes in the dataset while the others may have bad results. Hence, we have achieved some acceptable results but they can be improved and we will give some ideas in the next chapter.

Chapter 6
Conclusion and Future Work

This chapter summarises the methods we have implemented and also gives a brief description of the potential improvements that may be taken into consideration in future works.

6.1 Conclusion

In conclusion, this thesis has focused on the impact of adversarial attacks on deep learning algorithms in the health area, and ways to mitigate these attacks. In the chapter.2 "Related Work" we have described many adversarial attacks and countermeasures that exist in the literature as well as different datasets. To achieve our results we have decided to focus our work on only one dataset, the ISIC 2018, and only two adversarial attacks and mitigation methods (e.g. FGSM, UAP and adversarial retraining, LLT).

Firstly we have developed two operational skin lesion classifiers with the Inception V3 and ResNet 50 DL models. The following three experiments focused on attacking the previously trained models. In the first one, we performed a poisoning attack by modifying the training dataset, which allowed us to get a first approach of attacking a model. For the next two adversarial attack experiments, we have performed evasion attacks by adding noise into our input images with two methods, FGSM and UAP. With these adversarial attacks, we have demonstrated the capacity of attacking a DL model in healthcare. Finally, for the last experiments, we have focused on countermeasures. We have tested two methods: adversarial retraining and the use of LLT. Moreover, we have proposed a mitigation method by combining these two mitigation methods as a viable countermeasure against FGSM and UAP.

Since there is no real evidence in the literature focusing on the same dataset and the same adversarial attacks or countermeasures. The findings in this thesis, even though they are acceptable, could be considered exploratory.

Finally, we have conducted many experiments with the tools we have chosen, although we did not widen our pool of datasets or try more attacks due to the timing. Thus, we provide an operational base for potential future work.

6.2 Future work

As we have discussed in the chapter.5 "Discussion", the results of our methods are globally significant. We have achieved to answer our three research questions. However, there is still room for improvement and we have thought about some items.

Firstly, we would recommend experimenting with the methods we have used with other datasets and models to ensure the viability of such methods. Moreover, to put our methods into practice, it would require some modifications because attackers do not only threaten models by using Fast Gradient Sign Method or Universal Adversarial Perturbations. Hence, the exclusive use of our methods would not secure a model classifier in a real situation. Thus, it would also be interesting to analyze the performances of our mitigation method against other attacks mentioned in the chapter.2 "Related Work".

Secondly, if we look at our results, some metrics were poor due to the data imbalance in the dataset. Data imbalance is a problem in the health domain. Nevertheless, to improve the overall performances we would recommend the use of a more balanced dataset.

Thirdly, as we mentioned in the chapter.3 "Methodology", we wanted the countermeasures we have used to be mainly data-oriented. Thus, it would be interesting to compare the improvement we have made with our method combined with other countermeasures (e.g. model modification or additional model).

Finally, we also thought it would be interesting for future work to ask the medical corps their opinion on these results if they have any idea of methods to mitigate adversarial attacks by using medical knowledge.

References

[AMAZ17] Saad Albawi, Tareq Abed Mohammed, and Saad Al-Zawi. Understanding of a convolutional neural network. In *2017 International Conference on Engineering and Technology (ICET)*, pages 1–6, 2017.

[CPM20] Asma Channa, Nirvana Popescu, and Najeeb ur Rehman Malik. Robust technique to detect covid-19 using chest x-ray images. In *2020 International Conference on e-Health and Bioengineering (EHB)*, pages 1–6, 2020.

[CW17] Nicholas Carlini and David Wagner. Towards evaluating the robustness of neural networks, 2017.

[CYY20] Ferhat Ozgur Catak and Sule Yildirim Yayilgan. Deep neural network based malicious network activity detection under adversarial machine learning attacks. 06 2020.

[DYYH19] Binu Melit Devassy, Sule Yildirim-Yayilgan, and Jon Yngve Hardeberg. The impact of replacing complex hand-crafted features with standard features for melanoma classification using both hand-crafted and deep features. In Kohei Arai, Supriya Kapoor, and Rahul Bhatia, editors, *Intelligent Systems and Applications*, pages 150–159, Cham, 2019. Springer International Publishing.

[FZR$^+$19] Patrizia Ferroni, Fabio Massimo Zanzotto, Silvia Riondino, Noemi Scarpato, Fiorella Guadagni, and Mario Roselli. Breast cancer prognosis using a machine learning approach. *Cancers*, 11:328, 03 2019.

[GBV20] Margherita Grandini, Enrico Bagli, and Giorgio Visani. Metrics for multi-class classification: an overview, 2020.

[GKPB20] Divya Gopinath, Guy Katz, Corina S. Pasareanu, and Clark Barrett. Deepsafe: A data-driven approach for checking adversarial robustness in neural networks, 2020.

[Gon16] L. Gondara. Medical image denoising using convolutional denoising autoencoders. In *2016 IEEE 16th International Conference on Data Mining Workshops (ICDMW)*, pages 241–246, 2016.

[GPAM+14] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks, 2014.

[GSS15] Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples, 2015.

[HAA16] Mahmoud Hassaballah, Abdelmgeid Ali, and Hammam Alshazly. *Image Features Detection, Description and Matching*, volume 630, pages 11–45. 02 2016.

[HMT20] Hokuto Hirano, Akinori Minagi, and Kazuhiro Takemoto. Universal adversarial attacks on deep neural networks for medical image classification. 09 2020.

[HVD15] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network, 2015.

[HZRS16] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. *IEEE Conf. Comp. Vis. Patt. Recogn*, 2016.

[ISI] International skin imaging collaboration. https://challenge.isic-archive.com/.

[KGC+18] Daniel S. Kermany, Michael Goldbaum, Wenjia Cai, Carolina C.S. Valentim, Huiying Liang, Sally L. Baxter, Alex McKeown, Ge Yang, Xiaokang Wu, Fangbing Yan, Justin Dong, Made K. Prasadha, Jacqueline Pei, Magdalene Y.L. Ting, Jie Zhu, Christina Li, Sierra Hewett, Jason Dong, Ian Ziyar, Alexander Shi, Runze Zhang, Lianghong Zheng, Rui Hou, William Shi, Xin Fu, Yaou Duan, Viet A.N. Huu, Cindy Wen, Edward D. Zhang, Charlotte L. Zhang, Oulan Li, Xiaobo Wang, Michael A. Singer, Xiaodong Sun, Jie Xu, Ali Tafreshi, M. Anthony Lewis, Huimin Xia, and Kang Zhang. Identifying medical diagnoses and treatable diseases by image-based deep learning. *Cell*, 172(5):1122–1131.e9, 2018.

[KH15] Shaoqing Ren Jian Sun Kaiming He, Xiangyu Zhang. Deep residual learning for image recognition. *Microsoft Research*, 2015.

[KSH12] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 25. Curran Associates, Inc., 2012.

[LDJ+20] Jiannan Liu, Chuanpeng Dong, Guanglong Jiang, Xiaoyu Lu, Yunlong Liu, and Huanmei Wu. Transcription factor expression as a predictor of colon cancer prognosis: a machine learning practice. *BMC medical genomics*, 13(Suppl 9):135–135, 2020.

[LIF17] Jiajun Lu, Theerasit Issaranon, and David Forsyth. Safetynet: Detecting and rejecting adversarial examples robustly, 2017.

[LL18] Katherine M. Li and Evelyn C. Li. Skin lesion analysis towards melanoma detection via end-to-end deep learning of convolutional neural networks, 2018.

[LZL$^+$20] H. Li, Y. Zeng, G. Li, L. Lin, and Y. Yu. Online alternate generator against adversarial attacks. *IEEE Transactions on Image Processing*, 29:9305–9315, 2020.

[MDL$^+$20] L. Meng, D. Dong, L. Li, M. Niu, Y. Bai, M. Wang, X. Qiu, Y. Zha, and J. Tian. A deep learning prognosis model help alert for covid-19 patients at high-risk of death: A multi-center study. *IEEE Journal of Biomedical and Health Informatics*, pages 1–1, 2020.

[MFFF17] S. Moosavi-Dezfooli, A. Fawzi, O. Fawzi, and P. Frossard. Universal adversarial perturbations. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 86–94, 2017.

[MKSKRJ15] Mehran Mozaffari-Kermani, Susmita Sur-Kolay, Anand Raghunathan, and Niraj K. Jha. Systematic poisoning attacks on and defenses for machine learning in healthcare. *IEEE Journal of Biomedical and Health Informatics*, 19(6):1893–1905, 2015.

[MNG$^+$20] Xingjun Ma, Yuhao Niu, Lin Gu, Yisen Wang, Yitian Zhao, James Bailey, and Feng Lu. Understanding adversarial attacks on deep learning based medical image analysis systems, 2020.

[MYYH16] Tomáš Majtner, Sule Yildirim-Yayilgan, and Jon Yngve Hardeberg. Efficient melanoma detection using texture-based rsurf features. In Aurélio Campilho and Fakhri Karray, editors, *Image Analysis and Recognition*, pages 30–37, Cham, 2016. Springer International Publishing.

[NST$^+$18] Maria-Irina Nicolae, Mathieu Sinn, Minh Ngoc Tran, Beat Buesser, Ambrish Rawat, Martin Wistuba, Valentina Zantedeschi, Nathalie Baracaldo, Bryant Chen, Heiko Ludwig, Ian Molloy, and Ben Edwards. Adversarial robustness toolbox v1.2.0. *CoRR*, 1807.01069, 2018.

[QM20] Maël Nedellec Quentin Mouret. Adversarial attacks against deep learning algorithms and mitigation methods. Project report in TTM4502, Department of Information Security and Communication Technology, NTNU – Norwegian University of Science and Technology, Dec. 2020.

[QQBAF20] Adnan Qayyum, Junaid Qadir, Muhammad Bilal, and Ala Al-Fuqaha. Secure and robust machine learning for healthcare: A survey, 2020.

[QUQAF20] Adnan Qayyum, Muhammad Usama, Junaid Qadir, and Ala Al-Fuqaha. Securing connected & autonomous vehicles: Challenges posed by adversarial machine learning and the way forward. *IEEE Communications Surveys & Tutorials*, 22(2):998–1026, 2020.

[RDV17] Andrew Slavin Ross and Finale Doshi-Velez. Improving the adversarial robustness and interpretability of deep neural networks by regularizing their input gradients, 2017.

[RHDV17] Andrew Slavin Ross, Michael C. Hughes, and Finale Doshi-Velez. Right for the right reasons: Training differentiable models by constraining their explanations, 2017.

[Rip96] Brian D. Ripley. *Pattern Recognition and Neural Networks.* Cambridge University Press, 1996.

[Sah18] Sabyasachi Sahoo. Residual blocks — building blocks of resnet. *Towards Data Sciences*, 2018.

[SG18] Gokula Krishnan Santhanam and Paulina Grnarova. Defending against adversarial attacks by leveraging an entire gan, 2018.

[SLJ$^+$14] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions, 2014.

[SVI$^+$15] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jonathon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision, 2015.

[Tha18] Alaa Tharwat. Classification assessment methods: a detailed tutorial. 08 2018.

[Tsc18] Philipp Tschandl. The HAM10000 dataset, a large collection of multi-source dermatoscopic images of common pigmented skin lesions, 2018.

[YWWJT20] Zhaoxia Yin, Hua Wang, Jie Wang, and Wenzhong Wang Jin Tang. Defense against adversarial attacks by low-level image transformations. *Wiley Periodicals LLC*, 35(1453-1466):9305–9315, 2020.

[ZQD$^+$20] Fuzhen Zhuang, Zhiyuan Qi, Keyu Duan, Dongbo Xi, Yongchun Zhu, Hengshu Zhu, Hui Xiong, and Qing He. A comprehensive survey on transfer learning, 2020.

[Zuo18] Chandler Zuo. Regularization effect of fast gradient sign method and its generalization, 2018.