

# **Image Analysis and Recognition using Deep Learning for Dermatological Skin Conditions**

**Charles A. Hulebak**

# Table of Contents

---

- 1. Project Background**
- 2. Brief Overview - Deep Learning and Convolutional Neural Networks**
- 3. Dataset and Classification**
- 4. Clinical Characterization**
- 5. Project Walkthrough: Programming and Tuning the CNN**
- 6. Results and Conclusion**
- 7. References**
- 8. Addenda**
- 9. Creative Commons**

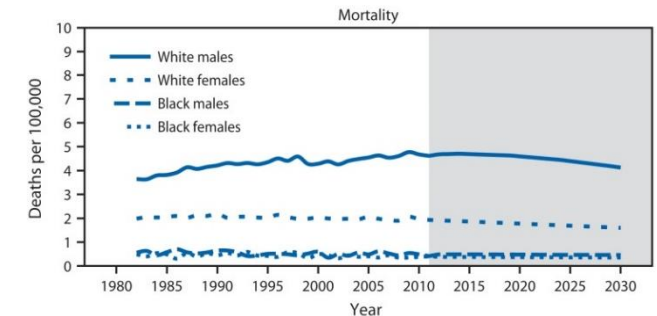
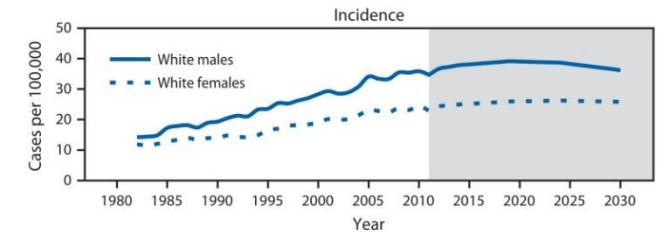
# Project Background

Skin cancer is the most common type of cancer worldwide and affects every race and gender.

Some of these unfortunate statistics in the United States include:

- Each year nearly **5 million people are treated** for all skin cancers combined, with more than **9,500 people diagnosed with skin cancer every day**, and more than **two people dying every hour**.<sup>1,2</sup>
- At least **one in five Americans** will develop skin cancer **by the age of 70**.<sup>3</sup>
- Basal cell carcinoma (BCC) is the most common form of skin cancer, with an estimated **4.3 million cases of BCC are diagnosed in the U.S. each year**.<sup>4</sup>
- The estimated **annual cost of treatment is estimated at \$4.8 billion in the U.S.**, with approximately \$3.3 billion dollars associated with costs attributable to melanoma.<sup>5</sup>
- **Globally, malignant skin melanoma has increased by 104.3%**, and non-melanoma skin cancer by 211.2% between 1990-2007, and 32.3% and 32.7% between 2007-2017.<sup>6</sup>

**Observed and projected age-adjusted melanoma incidence and mortality rates, by sex and race — United States, 1982–2030<sup>1</sup>**



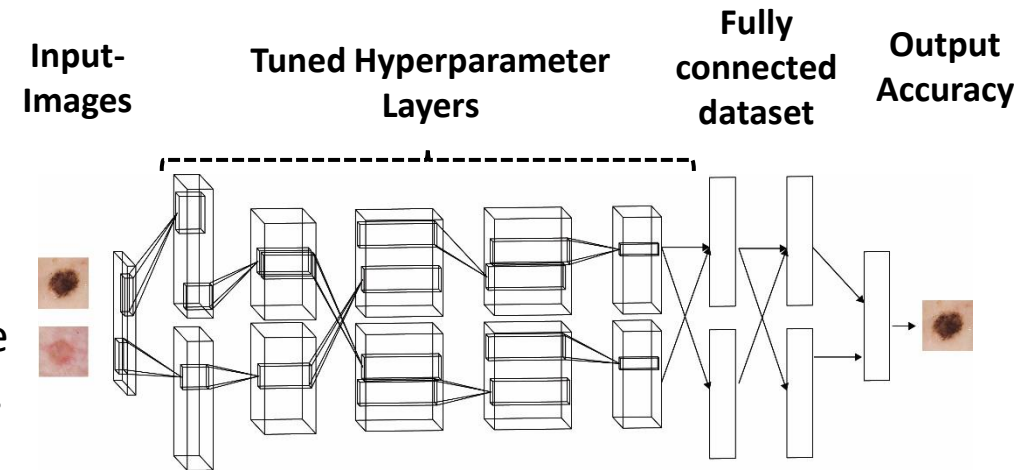
# Project Background

---

The use of technology, and specifically machine learning provides the ability to achieve these goals. Machine Learning, deep learning and innovations with convolutional neural networks (CNNs) provide the ability to analyze imagery data, classify, and provide results from inputted data. CNNs require large datasets in order to improve accuracy and provide sensible information.

The practical application of a machine learning program with a high success rate would provide the ability for increasing accurate results from early skin condition problems. Images of skin conditions could be tested with a reliable model, compared and have a probable diagnosis. A healthcare professional could take analyze the data, take this information into account, and treat conditions accordingly.

This project has the ability to influence further development of an initiative to decrease the amount of time required for early analysis of skin conditions and guide healthcare professionals, while increasing the response time for more serious conditions, and reducing the time invested for treating less serious conditions.



# Brief Overview - Deep Learning and Convolutionary Neural Networks

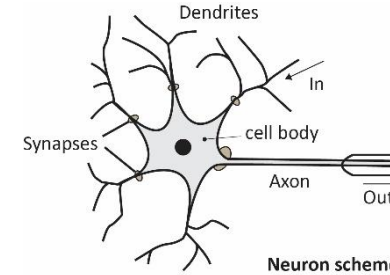
Deep learning is a subset of machine learning, which reflects the human learning process and derived from the biological processing of information. Convolutional neural networks (CNN) are a category of machine learning, in which unique features from a dataset are learned using filtered layers and a backpropagation algorithm to continually learn information about a dataset.

The process of convolution, in which CNNs obtain their name, is analogous to a filter (or stencil) passing over an image which is analyzed (by an activation map). Information is learned and passed to the subsequent layer for further processing, in which several layers may be introduced to more effectively learn information about the dataset.

Inputs are introduced with a dataset, which is trained (from layers of programmed code) to make accurate predictions. CNNs are regularized versions of multilayer perceptrons, which are fully connected neuron networks.

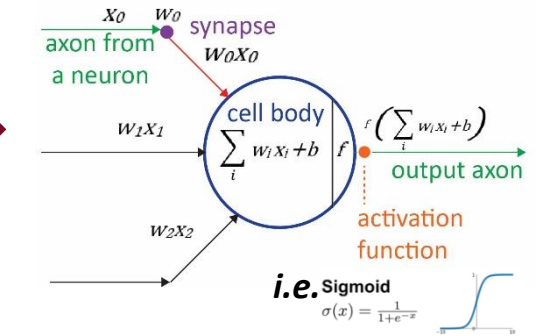
## Activation Function

### Biological Process Diagram



Images recreated from Stanford CS230 course

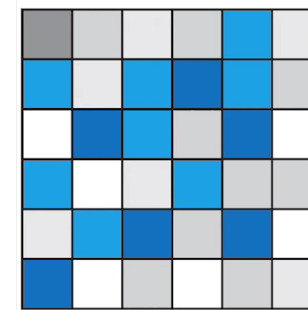
### Logical Process Diagram



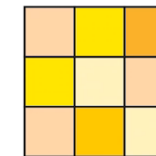
## Convolutional Layer

The image is passed over all of the spatial locations as it is passed along to the subsequent layer(s) toward the output.

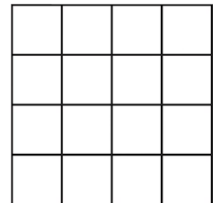
### Input Layer



### Hidden Layers



### Output Layers



# Brief Overview - Deep Learning and Convolutionary Neural Networks

---

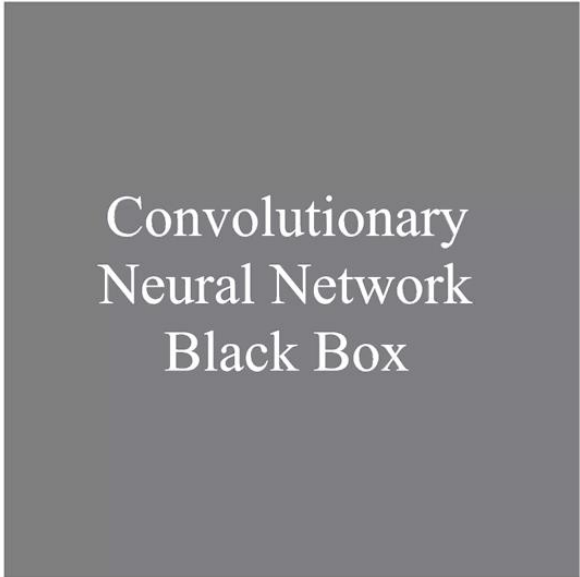
A series of layers are introduced to create a hierarchical structure, in which each layer (controlled by tuning hyperparameters) introduces various filters and increases a level of complexity to analyze the data.

Depending upon the data, and which layers are tested, accuracy may be determined from the test, training, and possible validation set (training and validation terminology may be interchanged). The measurement of accuracy reflects the machine's ability to learn from the data.

A series of epochs can test the data, while increasing the epochs allows the learning algorithm to run until errors have been minimized.

The accuracy of the model is tested and the hyperparameters are tuned for performance.

CNN 3D Walkthrough (animated, press play)



Convolutionary  
Neural Network  
Black Box



# Dataset and Classification

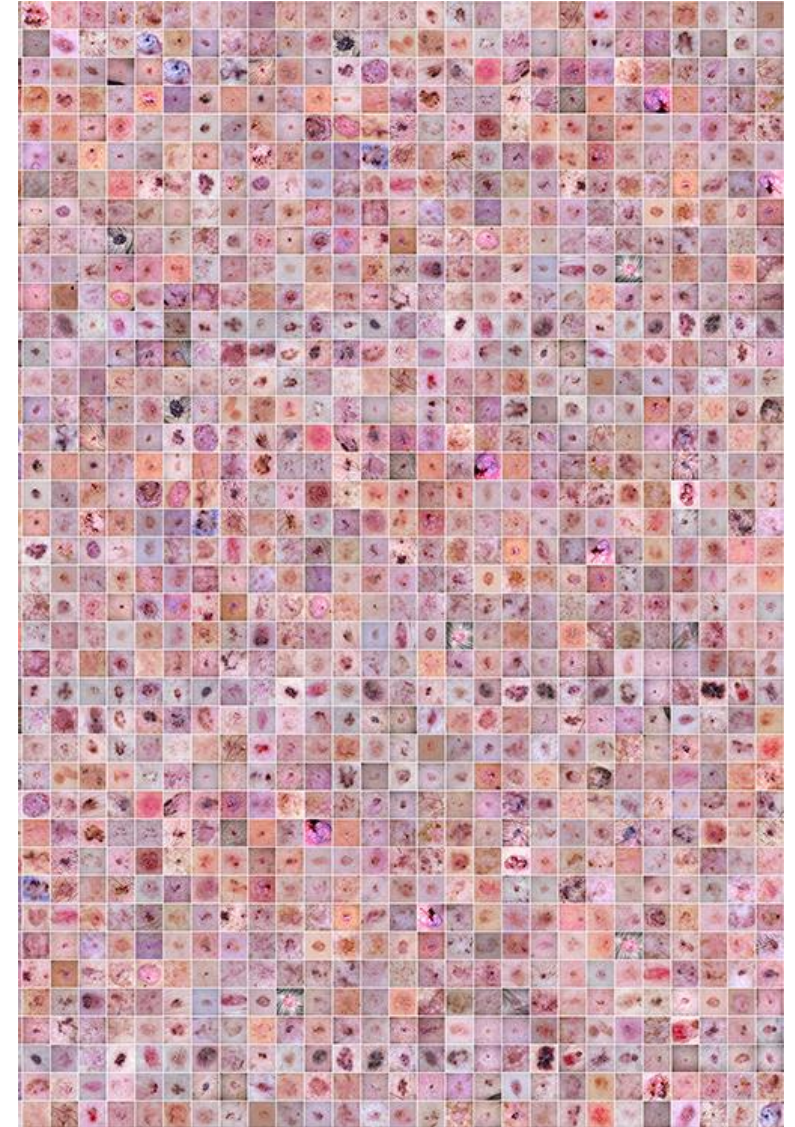
---

Innovations and initiatives are taking place around the world to address a multitude of healthcare concerns using data science, machine learning, and technology.

This project includes performing machine learning algorithms for image recognition. Utilizing a subset of over 10,000 images from Harvard's Dataverse, and also part of the Kaggle "Skin Cancer MNIST: HAM10000 analysis" competition.<sup>7</sup>

The dataset includes 10,000 images which contain seven categories of skin conditions. Although the CNN is capable of classifying the accuracy between each of these types of conditions, this project will focus on two specific skin conditions, including the *melanocytic nevi*, also called *nevi* (referred to as *mole* furthermore), and *basal cell carcinoma*, which is a form of cancerous cell.

The physical appearance of these two types of skin conditions have many similarities, however the basal cell carcinoma requires immediate attention for clinical treatment.



# Clinical Characterization

The clinical characterization of these two types of cells include:

## Mole characteristics <sup>8</sup>

- Size: generally less than 5mm diameter, pigmented spots on the skin that usually appear between birth and childhood.
- Shape: round or oval
- Color: brown is the most common; may be tan, black, red, pink, blue, skin-toned, or colorless
- Characteristics: flat or slightly raised

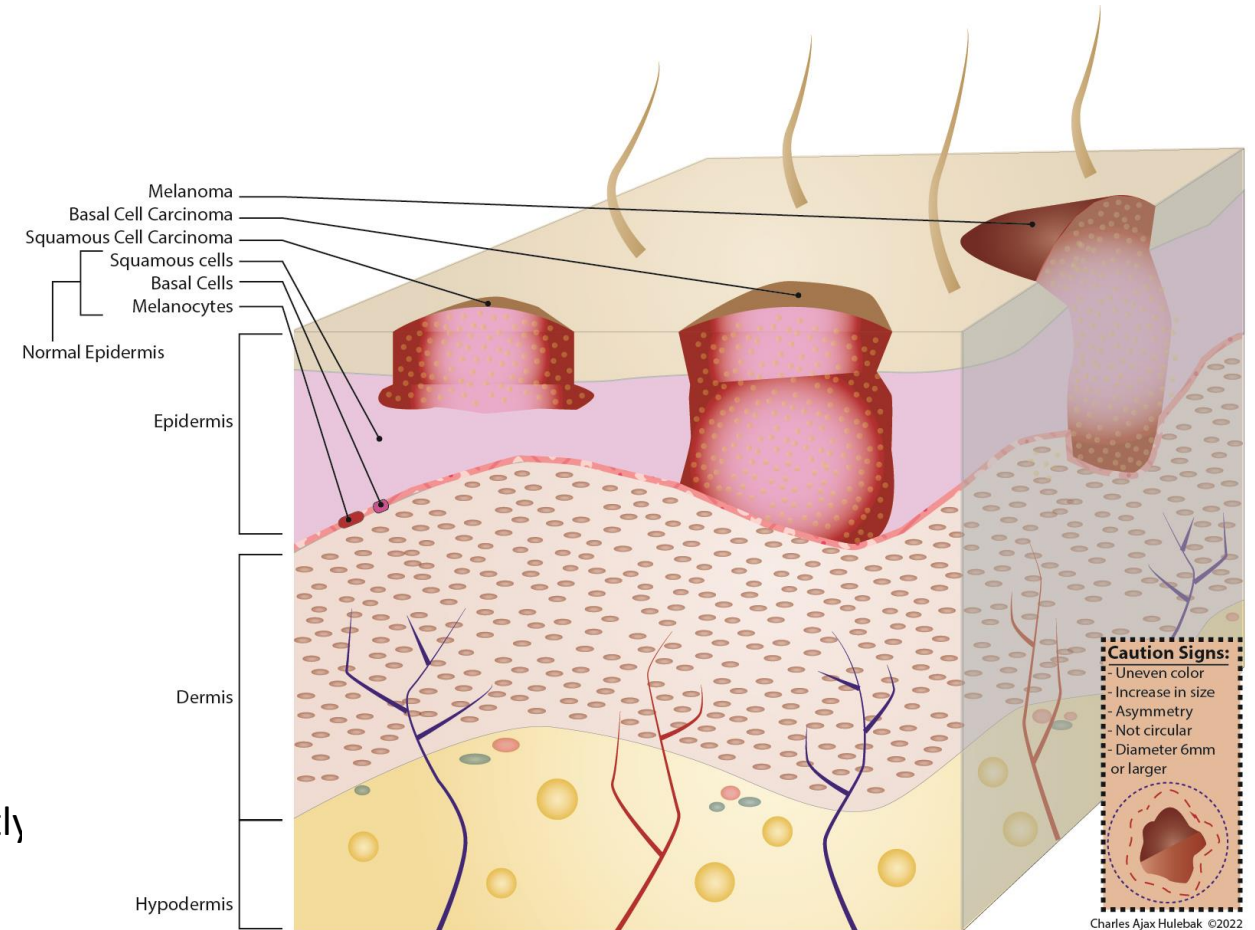
## Basal Cell Carcinoma characteristics <sup>9</sup>

- Size: varies
- Shape: varies
- Color: pink or red (most common); brown, black or flecks of these colors; yellowish; white; blue
- Characteristics: may be waxy, dip in the center, scaly patch of skin, sore that may bleed, ooze, or crust over

\*There are several stages of this skin condition, which have slightly different characteristics.

There are many similar characteristics, yet distinct differences. How is a deep learning program going to distinguish between them?

Skin Cells Diagram





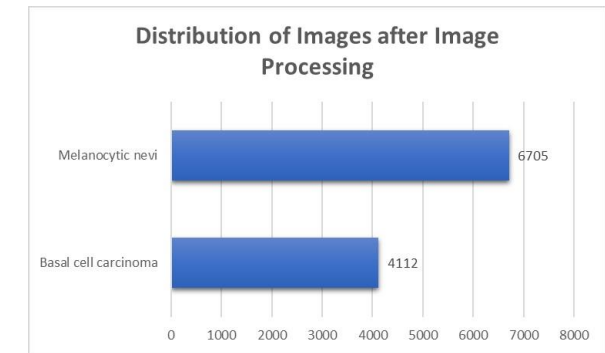
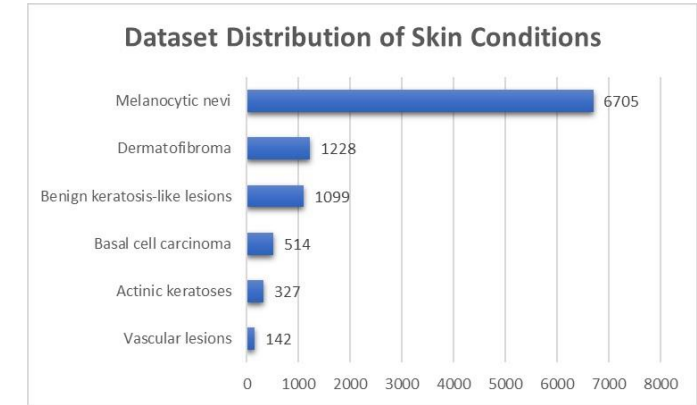
# Project Walkthrough: Programming and Tuning the CNN

The initial steps for this project included downloading and reviewing both the metadata and image data for the project.

Following the analysis of the data and determination of the project objective, the intent was to compare two variables, much like the “Cats vs. Dogs” project. In this case, we are assessing the difference between two types of skin cells that are difficult to distinguish with the eye.

The original dataset included 6,705 mole images, and 514 basal cell carcinoma images. Although this was sufficient for preliminary testing, I decided to create more basal cell carcinoma images by rotating and mirroring these images, to create 4,112 images. Increasing this volume of data provides the ability to generate more accurate predictions for the testing and training data. The new combined testing data is 10,817 images.

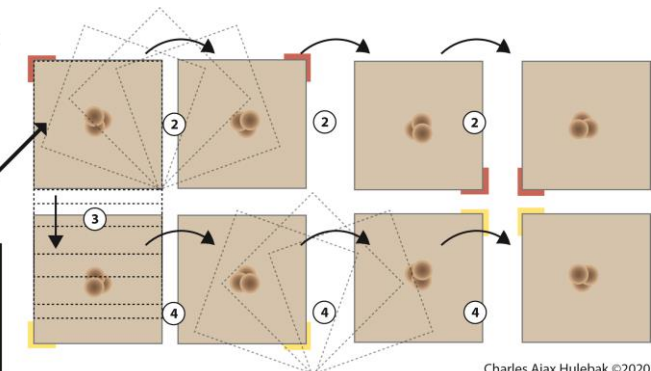
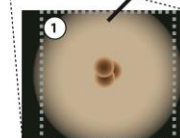
In addition, the image sizes were very large, amounting to over 2GB of data and lengthy initial testing. The images were reduced from 650px x 450 px at 300dpi to 300px \* 225px at 200dpi.



## Image preprocessing

Image preprocessing, and creating more image data for analysis:

1. Resampling and cropping original image
  2. Rotating images
  3. Mirroring/Reflecting images
  4. Rotating again
- 1 image = 8 images for analysis



Charles Ajax Hulebak ©2020

Charles A. Hulebak

# Project Walkthrough: Programming and Tuning the CNN

---

The Cats vs. Dogs project provide a great baseline for testing various features within the CNN. I imported programming libraries that were necessary to conduct analysis for the project, including Pandas for data processing, NumPy for linear algebra, the Scikit-image package, TensorFlow, Keras, and a few others shown within the code section.

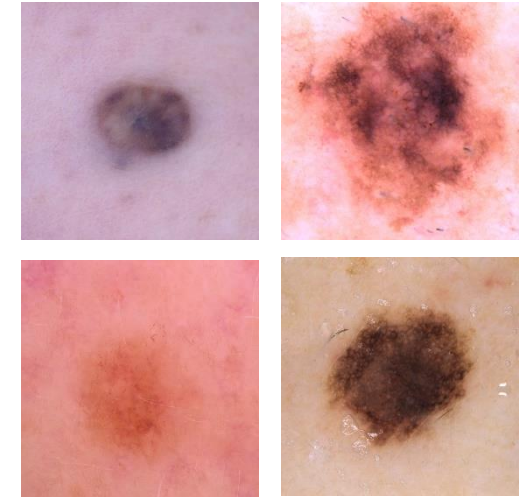
The initial testing results were significantly higher for early models, and it was evident that greater dpi resulted in higher accuracy for models. For example, the image dataset which contained images that were 450px x 450px at 300dpi had initial results around 85% with a single epoch.

Whereas this same image dataset that was reduced to 225px x 225px at 300 dpi were testing with a 65% accuracy rate.

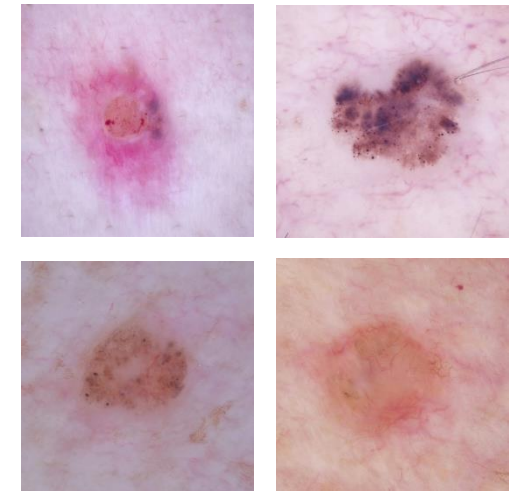
A third dataset was used with smaller images, and resulted with an average of a 55% accuracy rating.

The time required to test the larger dataset was also significantly longer, and I was more interested in testing more models for performance than reaching a very high rating on the large dataset (which might be considered in the future).

**Mole Images**



**Basal Cell Carcinoma Images**



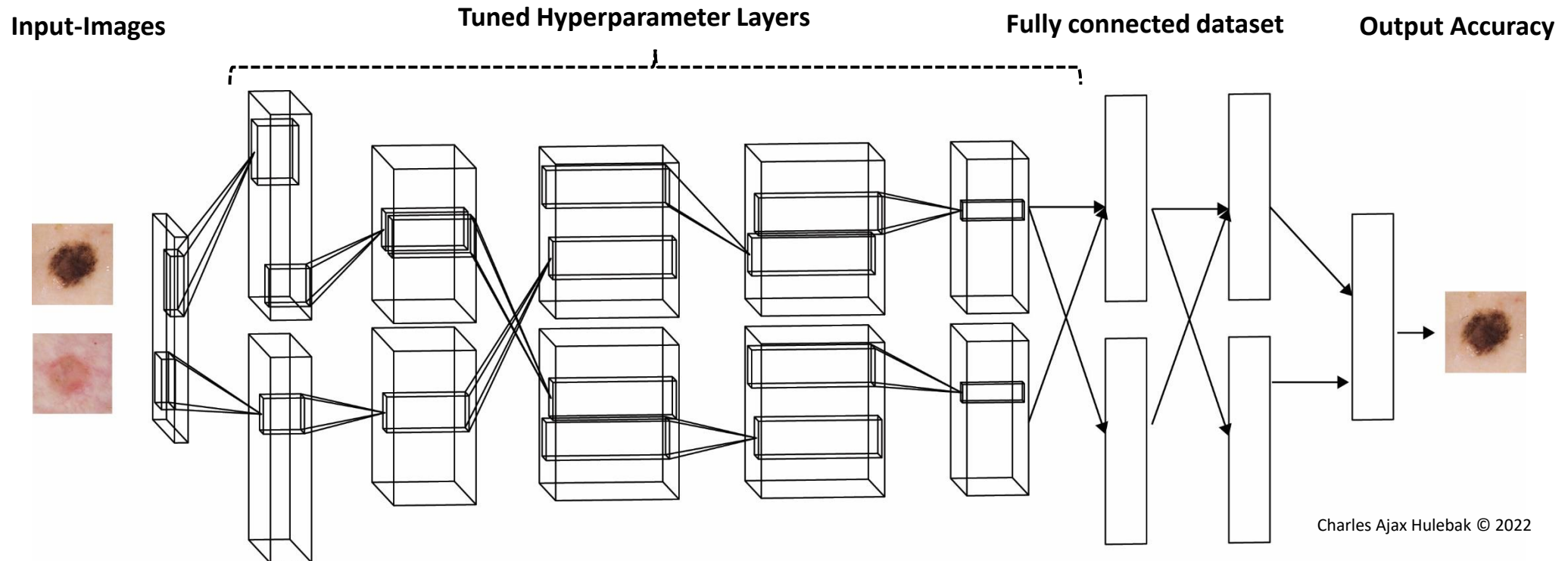
*Image Credit: HAM10000 dataset*

*Charles A. Hulebak*

# Project Walkthrough: Programming and Tuning the CNN

Through the process of modifying layer filters and tuning hyperparameters, I initially tested the model to determine the highest accuracy rate on a single epoch.

I continued to tune various layers and chose the best performing model within a single epoch. Then tested these on two epochs, followed by selecting the best performing model for further analysis. This model was tested with 10 epochs and evaluate the test accuracy, which resulted in 92.15% for the test data. It was evident that the learning process during these epochs provided much better results into this fully connected machine learning network.



# Results and Conclusion

I am very satisfied with the results from this project, as well as confident that the accuracy rating can be increased by analyzing higher resolution photographs, as well as increasing the sample size.

This project has a lot of potential, as well as other affiliated scientific and technology applications.

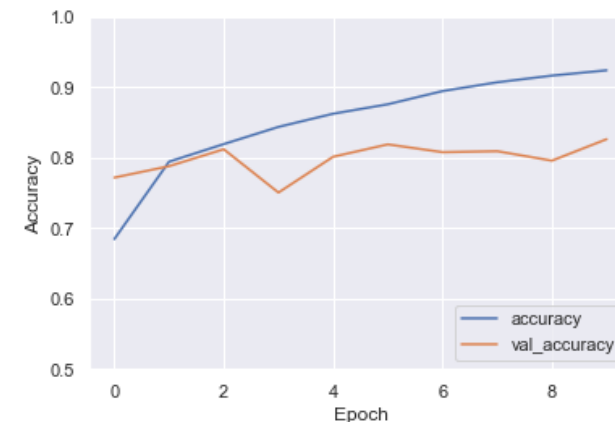
I was originally inspired to perform this project in light of the current COVID-19 pandemic, which is a global challenge and being addressed from many different perspectives to contain this virus, reduce the spread, and treat the those that have been infected. How can technology improve and control a condition such as this?

The CNN for this project has the ability to analyze a multitude of imagery, ranging from X-rays, to biological matter. Analysis can be conducted much faster for detection and containing than some of the other methods currently available.

I am certain that machine learning will continue to positively influence our future.

## Testing and Training Data Results

```
Train on 8653 samples, validate on 2164 samples
Epoch 1/10
8653/8653 [=====] - 1353s 156ms/sample - loss: 22.9174 - accuracy: 0.6845
- val_loss: 0.4868 - val_accuracy: 0.7717
Epoch 2/10
8653/8653 [=====] - 1329s 154ms/sample - loss: 0.4605 - accuracy: 0.7943
- val_loss: 0.4902 - val_accuracy: 0.7879
Epoch 3/10
8653/8653 [=====] - 1343s 155ms/sample - loss: 0.4104 - accuracy: 0.8193
- val_loss: 0.4831 - val_accuracy: 0.8119
Epoch 4/10
8653/8653 [=====] - 1352s 156ms/sample - loss: 0.3609 - accuracy: 0.8436
- val_loss: 0.5081 - val_accuracy: 0.7505
Epoch 5/10
8653/8653 [=====] - 1300s 150ms/sample - loss: 0.3224 - accuracy: 0.8624
- val_loss: 0.4634 - val_accuracy: 0.8013
Epoch 6/10
8653/8653 [=====] - 1316s 152ms/sample - loss: 0.2949 - accuracy: 0.8758
- val_loss: 0.4709 - val_accuracy: 0.8189
Epoch 7/10
8653/8653 [=====] - 1335s 154ms/sample - loss: 0.2599 - accuracy: 0.8945
- val_loss: 0.5123 - val_accuracy: 0.8078
Epoch 8/10
8653/8653 [=====] - 1358s 157ms/sample - loss: 0.2305 - accuracy: 0.9070
- val_loss: 0.4601 - val_accuracy: 0.8091
Epoch 9/10
8653/8653 [=====] - 1395s 161ms/sample - loss: 0.2118 - accuracy: 0.9164
- val_loss: 0.5344 - val_accuracy: 0.7957
Epoch 10/10
8653/8653 [=====] - 1375s 159ms/sample - loss: 0.2015 - accuracy: 0.9240
- val_loss: 0.5419 - val_accuracy: 0.8262
```



# References

---

1. Guy, G. P., Jr, Thomas, C. C., Thompson, T., Watson, M., Massetti, G. M., Richardson, L. C., & Centers for Disease Control and Prevention (CDC) (2015). Vital signs: melanoma incidence and mortality trends and projections - United States, 1982-2030. MMWR. Morbidity and mortality weekly report, 64(21), 591–596.
2. Rogers HW, Weinstock MA, Harris AR, et al. "Incidence Estimate of Nonmelanoma Skin Cancer in the United States", Arch Dermatol. 2010;146(3):283–287. (online) Available: <https://doi:10.1001/archdermatol.2010.19>
3. "Skin Cancer Facts & Statistics - The Skin Cancer Foundation". (2020). (online) Available: <https://www.skincancer.org/skin-cancer-information/skin-cancer-facts/>
4. "What are basal and squamous cell skin cancers?" American Cancer Society. (online) Available: <http://www.cancer.org/cancer/skincancer-basalandsquamouscell/detailedguide/skin-cancer-basal-and-squamous-cell-what-is-basal-and-squamous-cell>.
5. "Medical Expenditure Panel Survey", Agency for Healthcare Research and Quality, 2022. Rockville, MD: Agency for Healthcare Research and Quality; [January 2014]. (online) Available: <https://www.ahrq.gov/data/meps.html>
6. James, S., Abate, D., Abate, K., Abay, S., Abbafati, C., Abbasi, N. et al. (2018). "Global, regional, and national incidence, prevalence, and years lived with disability for 354 diseases and injuries for 195 countries and territories, 1990–2017: a systematic analysis for the Global Burden of Disease Study 2017", The Lancet, 392(10159), 1789-1858. doi: 10.1016/S0140-6736(18)32279-7
7. Tschandl, Philipp, 2018, "The HAM10000 dataset, a large collection of multi-source dermoscopic images of common pigmented skin lesions", <https://doi.org/10.7910/DVN/DBW86T>, Harvard Dataverse, V3, UNF:6:/APKSsDGVDhwPBWzsStU5A== [fileUNF] (online) Available: <https://dataverse.harvard.edu/dataset.xhtml?persistentId=doi:10.7910/DVN/DBW86T>
8. "Moles: Signs and symptoms", American Academy of Dermatology Association, 2022. (online) Available: <https://www.aad.org/public/diseases/a-z/moles-symptoms>
9. "Basal Cell Carcinoma: Overview", American Academy of Dermatology Association, 2022. (online) Available: <https://www.aad.org/public/diseases/skin-cancer/types/common/bcc>



# Addendum A - Global Prevalence of Skin Conditions

	Prevalence (thousands) 2017 counts	Incidence (thousands) 2017 counts	YLDs (thousands)				
			2017 counts	Percentage change in counts, 1990–2007	Percentage change in counts, 2007–17	Percentage change in age-standardised rates, 1990–2007	Percentage change in age-standardised rates, 2007–17
	7 369 526·2 (7 344 769·0 to 7 392 430·8)	38 480 253·2 (36 469 390·1 to 40 567 963·0)	853 042·6 (642 084·6 to 1 097 347·2)	29·8% (28·8 to 30·8)	17·0% (16·4 to 17·6)	–3·0% (–3·5 to –2·5)	–0·9% (–1·4 to –0·4)
Malignant skin melanoma	2324·4 (1794·8 to 2796·2)	308·7 (237·6 to 365·9)	140·9 (90·8 to 201·6)	104·3% (82·1 to 111·6)	32·3% (27·0 to 36·9)	41·3% (24·4 to 46·9)	4·9% (0·9 to 8·6)
Diagnosis and primary therapy phase of malignant skin melanoma	64·1 (49·0 to 76·8)	308·7 (237·6 to 365·9)	17·5 (11·0 to 25·2)	110·3% (89·4 to 118·3)	32·4% (26·1 to 38·8)	46·4% (30·5 to 52·6)	5·4% (0·5 to 10·6)
Controlled phase of malignant skin melanoma	2208·7 (1709·0 to 2662·4)	..	101·2 (59·9 to 154·2)	113·8% (93·9 to 121·4)	32·9% (27·7 to 37·9)	50·2% (34·9 to 55·4)	6·4% (2·3 to 10·7)
Metastatic phase of malignant	44·3 (32·7 to 49·6)	..	18·4 (11·3 to 25·1)	68·4% (40·0 to 80·0)	29·5% (22·6 to 35·5)	12·5% (–7·3 to 20·9)	–1·6% (–6·9 to 3·2)
Terminal phase of malignant skin	7·3 (5·4 to 8·2)	..	3·9 (2·4 to 5·2)	64·2% (37·2 to 73·7)	28·7% (21·6 to 33·9)	9·0% (–9·7 to 15·6)	–2·6% (–7·8 to 1·3)
Non-melanoma skin cancer	2537·1 (1666·4 to 3696·8)	7663·6 (5251·1 to 10 570·3)	90·2 (49·5 to 149·0)	211·2% (142·3 to 305·0)	32·7% (25·3 to 40·0)	96·8% (46·7 to 162·7)	–2·0% (–7·7 to 3·9)
Non-melanoma skin cancer (squamous-cell carcinoma)	2158·9 (1294·8 to 3255·8)	1778·8 (1068·8 to 2620·9)	87·7 (46·9 to 146·3)	221·7% (148·5 to 334·5)	32·8% (25·0 to 40·0)	102·5% (47·3 to 180·3)	–2·0% (–7·8 to 4·0)
Non-melanoma skin cancer (basal-cell carcinoma)	596·8 (325·9 to 947·0)	5884·8 (3702·9 to 8742·9)	2·5 (0·9 to 5·2)	44·5% (20·3 to 76·5)	30·8% (23·1 to 37·0)	–3·8% (–21·1 to 18·1)	–1·0% (–6·4 to 3·2)
Skin and subcutaneous diseases	1 974 238·4 (1 916 671·8 to 2 034 645·7)	4 185 971·3 (3 971 760·5 to 4 391 218·2)	41 621·9 (27 371·7 to 61 859·5)	24·0% (22·9 to 25·3)	13·0% (12·5 to 13·6)	0·9% (0·4 to 1·4)	0·6% (0·1 to 1·1)
Dermatitis	291 689·4 (276 520·7 to 308 059·0)	274 034·1 (246 120·0 to 302 497·5)	11 128·1 (6484·1 to 17 733·4)	19·3% (18·2 to 20·7)	12·1% (11·3 to 12·8)	–0·1% (–0·7 to 0·5)	1·1% (0·4 to 1·8)
Atopic dermatitis	205 517·2 (195 701·2 to 218 634·4)	27 134·4 (25 282·9 to 29 055·0)	9003·4 (4887·0 to 14 981·0)	17·0% (16·1 to 17·9)	11·6% (10·8 to 12·5)	0·4% (–0·4 to 1·1)	1·7% (0·9 to 2·6)
Contact dermatitis	79 666·7 (70 425·7 to 89 554·9)	221 252·8 (193 775·3 to 249 230·6)	1989·2 (1304·4 to 2950·5)	31·0% (28·8 to 33·2)	14·4% (12·6 to 16·2)	–1·6% (–2·3 to –1·0)	–1·1% (–1·7 to –0·5)
Seborrheic dermatitis	10 035·9 (9450·2 to 10 668·4)	25 646·9 (23 981·6 to 27 315·9)	135·6 (78·1 to 215·0)	20·8% (18·5 to 23·3)	8·4% (6·6 to 10·3)	–7·7% (–9·0 to –6·4)	–7·1% (–8·3 to –5·8)
Psoriasis	64 609·6 (62 454·3 to 66 767·5)	7846·6 (7564·9 to 8162·8)	5569·5 (3956·1 to 7354·3)	43·1% (42·0 to 44·2)	21·1% (20·3 to 21·9)	5·1% (4·3 to 5·8)	2·3% (1·6 to 2·9)
Bacterial skin diseases	11 397·9 (11 061·4 to 11 741·7)	266 779·7 (260 229·3 to 273 665·3)	177·9 (112·3 to 274·1)	29·2% (27·2 to 31·0)	15·5% (14·0 to 17·1)	2·0% (–0·5 to 4·4)	0·8% (–1·1 to 2·9)
Cellulitis	2071·5 (1952·2 to 2186·2)	42 958·6 (40 535·7 to 45 172·9)	118·2 (78·7 to 167·8)	26·7% (24·9 to 28·6)	12·9% (11·7 to 14·2)	–2·6% (–3·8 to –1·3)	–2·9% (–3·8 to –2·0)
Pyoderma	10 557·2 (10 245·5 to 10 853·2)	223 821·2 (217 699·0 to 230 266·1)	59·6 (23·9 to 123·5)	35·0% (33·7 to 36·4)	20·9% (19·8 to 22·2)	13·0% (11·9 to 14·2)	8·8% (7·8 to 9·9)
Impetigo	4620·7 (4358·2 to 4870·5)	84 007·5 (79 485·3 to 88 529·9)	26·4 (10·4 to 55·8)	35·7% (33·6 to 37·8)	20·7% (18·7 to 22·7)	24·1% (22·2 to 26·0)	13·4% (11·5 to 15·2)
Abscess and other bacterial skin diseases	5936·4 (5770·4 to 6098·7)	139 813·6 (135 314·1 to 143 965·8)	33·2 (13·4 to 69·5)	34·5% (32·8 to 36·3)	21·1% (19·7 to 22·6)	5·5% (4·2 to 6·8)	5·1% (3·8 to 6·4)
Scabies	175 406·7 (154 517·9 to 198 404·1)	527 476·5 (462 050·9 to 598 087·9)	4528·7 (2506·4 to 7414·6)	16·1% (13·8 to 18·5)	6·6% (5·3 to 8·0)	–4·3% (–5·0 to –3·6)	–3·1% (–3·6 to –2·6)
Fungal skin diseases	743 458·4 (681 568·4 to 808 149·7)	2 126 927·9 (1 917 361·6 to 2 317 274·7)	4154·5 (1651·4 to 8633·2)	21·5% (19·0 to 24·0)	10·9% (9·2 to 12·5)	–3·1% (–3·8 to –2·4)	–4·4% (–5·4 to –3·4)
Tinea capitis	160 239·3 (133 390·6 to 194 433·3)	303 016·6 (245 340·4 to 369 760·2)	916·5 (357·3 to 1965·4)	–5·8% (–7·3 to –4·3)	–13·0% (–14·5 to –11·3)	–13·8% (–15·0 to –12·7)	–19·0% (–20·4 to –17·3)
Other fungal skin diseases	583 219·1 (526 500·8 to 645 363·9)	1 823 911·3 (1 638 297·9 to 2 009 788·9)	3238·0 (1305·4 to 6694·4)	37·1% (35·2 to 39·3)	20·2% (19·0 to 21·6)	1·8% (1·5 to 2·1)	1·3% (1·1 to 1·5)
Viral skin diseases	130 639·2 (125 604·0 to 136 047·9)	116 329·8 (111 012·4 to 121 710·1)	4033·0 (2595·4 to 5995·6)	10·4% (9·8 to 11·1)	6·4% (6·0 to 6·9)	–2·8% (–3·2 to –2·3)	–1·8% (–2·2 to –1·4)
Viral warts	54 309·6 (52 104·2 to 56 403·2)	30 140·5 (29 078·5 to 31 211·0)	1662·2 (1066·0 to 2439·8)	13·7% (12·8 to 14·6)	5·6% (4·8 to 6·3)	–6·8% (–7·3 to –6·3)	–4·8% (–5·4 to –4·3)
Molluscum contagiosum	76 329·6 (71 596·6 to 81 134·1)	86 189·3 (80 984·6 to 91 427·2)	2370·7 (1511·7 to 3525·9)	8·2% (7·5 to 8·9)	7·1% (6·5 to 7·6)	0·2% (–0·3 to 0·8)	0·3% (–0·3 to 0·8)
Acne vulgaris	119 082·3 (107 127·9 to 133 651·4)	60 118·8 (53 260·2 to 68 180·7)	2547·6 (1518·8 to 4056·6)	46·1% (44·6 to 47·6)	16·2% (15·2 to 17·2)	19·8% (18·6 to 20·9)	11·4% (10·3 to 12·5)
Alopecia areata	15 981·0 (15 477·3 to 16 515·9)	28 185·2 (27 302·2 to 29 126·3)	523·1 (334·9 to 774·8)	28·8% (27·6 to 30·1)	12·7% (11·8 to 13·8)	–2·3% (–3·1 to –1·4)	–1·5% (–2·3 to –0·6)
Pruritus	71 224·3 (63 948·2 to 80 034·2)	55 643·1 (49 208·3 to 62 689·4)	755·6 (356·1 to 1433·2)	36·4% (34·1 to 38·8)	18·9% (17·5 to 20·4)	2·9% (2·5 to 3·4)	1·6% (1·2 to 2·0)
Urticaria	83 610·0 (73 335·4 to 95 162·9)	147 198·5 (129 941·2 to 166 345·6)	5014·8 (3321·0 to 7046·4)	19·3% (17·4 to 21·5)	10·8% (9·8 to 11·9)	0·7% (0·2 to 1·2)	0·3% (–0·2 to 0·8)
Decubitus ulcer	1143·7 (1022·6 to 1288·5)	4199·3 (3752·4 to 4741·3)	181·2 (125·8 to 244·3)	45·2% (42·7 to 47·7)	28·9% (26·3 to 31·9)	–3·3% (–4·9 to –1·8)	–0·4% (–2·2 to 1·5)
Other skin and subcutaneous diseases	550 810·3 (538 490·7 to 563 961·3)	571 231·9 (558 726·3 to 584 649·5)	3008·1 (1446·7 to 5557·4)	45·1% (44·6 to 45·6)	25·6% (25·2 to 26·0)	6·6% (6·4 to 6·8)	4·3% (4·1 to 4·5)

Table: Global prevalence, incidence, and YLDs for 2017; percentage change of YLD counts; and percentage change of age-standardized YLD rates for 1990–2007 and 2007–17 for both sexes combined for all Level 5 causes, nature of injury aggregates, and nine impairments (abbreviated content)<sup>6</sup>

# Addendum B-Jupyter notebook excerpts (1)

## Image Analysis and Recognition using a CNN and Skin Condition Data

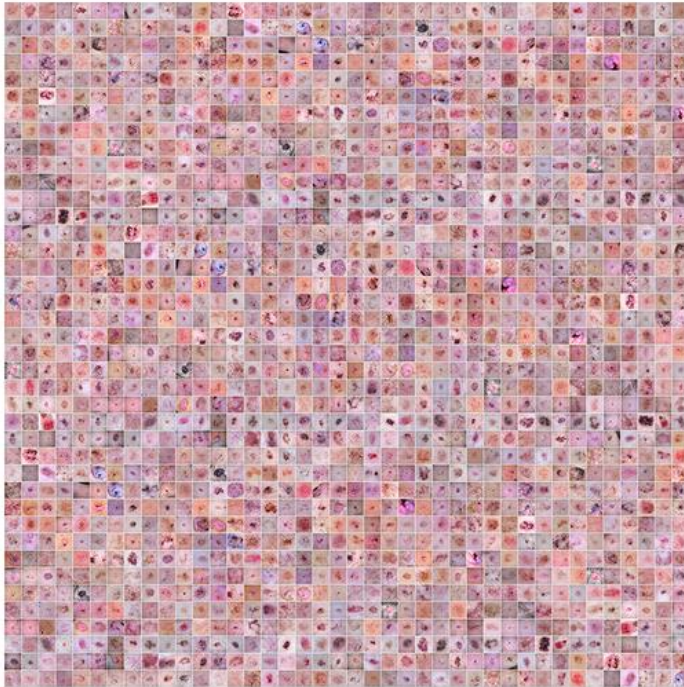
Introduction to Business Applications of Machine Learning (XIS22)

This project references the Kaggle Skin Cancer MNIST: HAM10000 analysis

Original Data Sources

-Original Challenge: <https://challenge2018.kaggle.com/https://paperswithcode.com/sota/skin-lesion-detection-on-ham10000/2018-07-10/2018-07-10>  
Noel Codella, Veronica Rotemberg, Philipp Tschandl, M. Emre Celebi, Stephen Dusza, David Guzman, Brian Helms, Aadi Kallio, Konstantinos Liopyris, Michael Marchetti, Harald Kittler, Allan Halpern: "Skin Lesion Analysis Toward Melanoma Detection 2018: A Challenge Hosted by the International Skin Imaging Collaboration (ISIC)", 2018, <https://arxiv.org/abs/1808.10383> (2) Tschandl, P., Rosendahl, C. & Kittler, H. The HAM10000 dataset, a large collection of multi-source dermatoscopic images of common pigmented skin lesions. Sci. Data 5, 180161 doi:10.1038/sdata.2018.161 (2018).

This project will take a subset of original 10,000 images for the specific classification of one of two skin conditions. These conditions, which can be easily mistaken from one another include the mole and signs of basal cell carcinoma. Utilizing a convolutional neural network (CNN), this machine learning project is seeking to identify physical characteristics from images to determine if an image has a high probability of being one or the other.



```
In [3]: 1 #Import libraries
2
3 import pandas as pd # data processing, CSV file
4
5 import numpy as np # linear algebra
6
7 #skikit-image package
8 import matplotlib.pyplot as plt
9 import matplotlib.image
10 import matplotlib.mpl as mlab
11 import matplotlib.patches as mpatches
12
13 import skimage.io
14 from skimage import data
15 from skimage.io import imread
16 from skimage.io import imread_collection
17 from skimage.io import imsave
18
19 import tensorflow as tf
20 from tensorflow.keras.layers import Flatten, Dense
21 import cv
22 import os
23 from os.path import join, exists, basename, normpath, split
24
25 import math
26 import random
27
28 #non-relative paths
29 import glob2
30 from glob2 import glob
31
32 import cv2
33 import imutils
34
35 from shutil import copyfile
36 from scipy.stats import norm
37 import random as rnd
38 rnd.seed(12345)
39
40 skin_analysis = os.path.join(".", "input")
```

Let's take a look at the original dataset:

```
In [2]: 1 imageid_path_dict = {os.path.splitext(os.path.basename(x))[0]: x
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
1001
1002
1003
1004
1005
1006
1007
1008
1009
1010
1011
1012
1013
1014
1015
1016
1017
1018
1019
1020
1021
1022
1023
1024
1025
1026
1027
1028
1029
1030
1031
1032
1033
1034
1035
1036
1037
1038
1039
1040
1041
1042
1043
1044
1045
1046
1047
1048
1049
1050
1051
1052
1053
1054
1055
1056
1057
1058
1059
1060
1061
1062
1063
1064
1065
1066
1067
1068
1069
1070
1071
1072
1073
1074
1075
1076
1077
1078
1079
1080
1081
1082
1083
1084
1085
1086
1087
1088
1089
1090
1091
1092
1093
1094
1095
1096
1097
1098
1099
1100
1101
1102
1103
1104
1105
1106
1107
1108
1109
1110
1111
1112
1113
1114
1115
1116
1117
1118
1119
1120
1121
1122
1123
1124
1125
1126
1127
1128
1129
1130
1131
1132
1133
1134
1135
1136
1137
1138
1139
1140
1141
1142
1143
1144
1145
1146
1147
1148
1149
1150
1151
1152
1153
1154
1155
1156
1157
1158
1159
1160
1161
1162
1163
1164
1165
1166
1167
1168
1169
1170
1171
1172
1173
1174
1175
1176
1177
1178
1179
1180
1181
1182
1183
1184
1185
1186
1187
1188
1189
1190
1191
1192
1193
1194
1195
1196
1197
1198
1199
1200
1201
1202
1203
1204
1205
1206
1207
1208
1209
1210
1211
1212
1213
1214
1215
1216
1217
1218
1219
1220
1221
1222
1223
1224
1225
1226
1227
1228
1229
1230
1231
1232
1233
1234
1235
1236
1237
1238
1239
1240
1241
1242
1243
1244
1245
1246
1247
1248
1249
1250
1251
1252
1253
1254
1255
1256
1257
1258
1259
1260
1261
1262
1263
1264
1265
1266
1267
1268
1269
1270
1271
1272
1273
1274
1275
1276
1277
1278
1279
1280
1281
1282
1283
1284
1285
1286
1287
1288
1289
1290
1291
1292
1293
1294
1295
1296
1297
1298
1299
1300
1301
1302
1303
1304
1305
1306
1307
1308
1309
1310
1311
1312
1313
1314
1315
1316
1317
1318
1319
1320
1321
1322
1323
1324
1325
1326
1327
1328
1329
1330
1331
1332
1333
1334
1335
1336
1337
1338
1339
1340
1341
1342
1343
1344
1345
1346
1347
1348
1349
1350
1351
1352
1353
1354
1355
1356
1357
1358
1359
1360
1361
1362
1363
1364
1365
1366
1367
1368
1369
1370
1371
1372
1373
1374
1375
1376
1377
1378
1379
1380
1381
1382
1383
1384
1385
1386
1387
1388
1389
1390
1391
1392
1393
1394
1395
1396
1397
1398
1399
1400
1401
1402
1403
1404
1405
1406
1407
1408
1409
1410
1411
1412
1413
1414
1415
1416
1417
1418
1419
1420
1421
1422
1423
1424
1425
1426
1427
1428
1429
1430
1431
1432
1433
1434
1435
1436
1437
1438
1439
1440
1441
1442
1443
1444
1445
1446
1447
1448
1449
1450
1451
1452
1453
1454
1455
1456
1457
1458
1459
1460
1461
1462
1463
1464
1465
1466
1467
1468
1469
1470
1471
1472
1473
1474
1475
1476
1477
1478
1479
1480
1481
1482
1483
1484
1485
1486
1487
1488
1489
1490
1491
1492
1493
1494
1495
1496
1497
1498
1499
1500
1501
1502
1503
1504
1505
1506
1507
1508
1509
1510
1511
1512
1513
1514
1515
1516
1517
1518
1519
1520
1521
1522
1523
1524
1525
1526
1527
1528
1529
1530
1531
1532
1533
1534
1535
1536
1537
1538
1539
1540
1541
1542
1543
1544
1545
1546
1547
1548
1549
1550
1551
1552
1553
1554
1555
1556
1557
1558
1559
1560
1561
1562
1563
1564
1565
1566
1567
1568
1569
1570
1571
1572
1573
1574
1575
1576
1577
1578
1579
1580
1581
1582
1583
1584
1585
1586
1587
1588
1589
1590
1591
1592
1593
1594
1595
1596
1597
1598
1599
1600
1601
1602
1603
1604
1605
1606
1607
1608
1609
1610
1611
1612
1613
1614
1615
1616
1617
1618
1619
1620
1621
1622
1623
1624
1625
1626
1627
1628
1629
1630
1631
1632
1633
1634
1635
1636
1637
1638
1639
1640
1641
1642
1643
1644
1645
1646
1647
1648
1649
1650
1651
1652
1653
1654
1655
1656
1657
1658
1659
1660
1661
1662
1663
1664
1665
1666
1667
1668
1669
1670
1671
1672
1673
1674
1675
1676
1677
1678
1679
1680
1681
1682
1683
1684
1685
1686
1687
1688
1689
1690
1691
1692
1693
1694
1695
1696
1697
1698
1699
1700
1701
1702
1703
1704
1705
1706
1707
1708
1709
1710
1711
1712
1713
1714
1715
1716
1717
1718
1719
1720
1721
1722
1723
1724
1725
1726
1727
1728
1729
1730
1731
1732
1733
1734
1735
1736
1737
1738
1739
1740
1741
1742
1743
1744
1745
1746
1747
1748
1749
1750
1751
1752
1753
1754
1755
1756
1757
1758
1759
1760
1761
1762
1763
1764
1765
1766
1767
1768
1769
1770
1771
1772
1773
1774
1775
1776
1777
1778
1779
1780
1781
1782
1783
1784
1785
1786
1787
1788
1789
1790
1791
1792
1793
1794
1795
1796
1797
1798
1799
1800
1801
1802
1803
1804
1805
1806
1807
1808
1809
1810
1811
1812
1813
1814
1815
1816
1817
1818
1819
1820
1821
1822
1823
1824
1825
1826
1827
1828
1829
1830
1831
1832
1833
1834
1835
1836
1837
1838
1839
1840
1841
1842
1843
1844
1845
1846
1847
1848
1849
1850
1851
1852
1853
1854
1855
1856
1857
1858
1859
1860
1861
1862
1863
1864
1865
1866
1867
1868
1869
1870
1871
1872
1873
1874
1875
1876
1877
1878
1879
1880
1881
1882
1883
1884
1885
1886
1887
1888
1889
1890
1891
1892
1893
1894
1895
1896
1897
1898
1899
1900
1901
1902
1903
1904
1905
1906
1907
1908
1909
1910
1911
1912
1913
1914
1915
1916
1917
1918
1919
1920
1921
1922
1923
1924
1925
1926
1927
1928
1929
1930
1931
1932
1933
1934
1935
1936
1937
1938
1939
1940
1941
1942
1943
1944
1945
1946
1947
1948
1949
1950
1951
1952
1953
1954
1955
1956
1957
1958
1959
1960
1961
1962
1963
1964
1965
1966
1967
1968
1969
1970
1971
1972
1973
1974
1975
1976
1977
1978
1979
1980
1981
1982
1983
1984
1985
1986
1987
1988
1989
1990
1991
1992
1993
1994
1995
1996
1997
1998
1999
2000
2001
2002
2003
2004
2005
2006
2007
2008
2009
2010
2011
2012
2013
2014
2015
2016
2017
2018
2019
2020
2021
2022
2023
2024
2025
2026
2027
2028
2029
2030
2031
2032
2033
2034
2035
2036
2037
2038
2039
2040
2041
2042
2043
2044
2045
2046
2047
2048
2049
2050
2051
2052
2053
2054
2055
2056
2057
2058
2059
2060
2061
2062
2063
2064
2065
2066
2067
2068
2069
2070
2071
2072
2073
2074
2075
2076
2077
2078
2079
2080
2081
2082
2083
2084
2085
2086
2087
2088
2089
2090
2091
2092
2093
2094
2095
2096
2097
2098
2099
2100
2101
2102
2103
2104
2105
2106
2107
2108
2109
2110
2111
2112
2113
2114
2115
2116
2117
2118
2119
2120
2121
2122
2123
2124
2125
2126
2127
2128
2129
2130
2131
2132
2133
2134
2135
2136
2137
2138
2139
2140
2141
2142
2143
2144
2145
2146
2147
2148
2149
2150
2151
2152
2153
2154
2155
2156
2157
2158
2159
2160
2161
2162
2163
2164
2165
2166
2167
2168
2169
2170
2171
2172
2173
2174
2175
2176
2177
2178
2179
2180
2181
2182
2183
2184
2185
2186
2187
2188
2189
2190
2191
2192
2193
2194
2195
2196
2197
2198
2199
2200
2201
2202
2203
2204
2205
2206
2207
2208
2209
2210
2211
2212
2213
2214
2215
2216
2217
2218
2219
2220
2221
2222
2223
2224
2225
2226
2227
2228
2229
2230
2231
2232
2233
2234
2235
2236
2237
2238
2239
2240
2241
2242
2243
2244
2245
2246
2247
2248
2249
2250
2251
2252
2253
2254
2255
2256
2257
2258
2259
2260
2261
2262
2263
2264
2265
2266
2267
2268
2269
2270
2271
2272
2273
2274
2275
2276
2277
2278
2279
2280
2281
2282
2283
2284
2285
2286
2287
2288
2289
2290
2291
2292
2293
2294
2295
2296
2297
2298
2299
2300
2301
2302
2303
2304
2305
2306
2307
2308
2309
2310
2311
2312
2313
2314
2315
2316
2317
2318
2319
2320
2321
2322
2323
2324
2325
2326
2327
2328
2329
2330
2331
2332
2333
2334
2335
2336
2337
2338
2339
2340
2341
2342
2343
2344
2345
2346
2347
2348
2349
2350
2351
2352
2353
2354
2355
2356
2357
2358
2359
2360
2361
2362
2363
2364
2365
2366
2367
2368
2369
2370
2371
2372
2373
2374
2375
2376
2377
2378
2379
2380
2381
2382
2383
2384
2385
2386
2387
2388
2389
2390
2391
2392
2393
2394
2395
2396
2397
2398
2399
2400
2401
2402
2403
2404
2405
2406
2407
2408
2409
2410
2411
2412
2413
2414
2415
2416
2417
2418
2419
2420
2421
2422
2423
2424
2425
2426
2427
2428
2429
2430
2431
2432
2433
2434
2435
2436
2437
2438
2439
2440
2441
2442
2443
2444
2445
2446
2447
2448
2449
2450
2451
2452
2453
2454
2455
2456
2457
2458
2459
2460
2461
2462
2463
2464
2465

```

# Addendum B-Jupyter notebook excerpts (2)

## Data Preprocessing

There are seven categories of skin conditions, however for this analysis we are interested in the ability to identify the differences between a common mole and a cancerous cell. These skin conditions are referred to as the melanocytic nevus (mole) and the basal cell carcinoma (cancerous cell). Let's take a look at these categories:

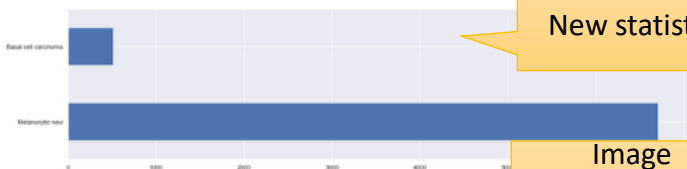
```
In [5]: 1 #Metadata information
2 title_dcf.describe(exclude=[np.number])
```

```
Out[5]:
```

	lesion_id	image_id	dx	dx_type	sex	localization	path	cell_type
count	7219	7219	7219	7219	7219	0	7219	7219
unique	5720	7219	2	3	3	15	0	2
top	HAVI_0003263	(SIC_0003212)	nv	follicular	male	back	hav	Melanocytic nevus
freq	6	1	8705	3704	3726	1613	hav	6705

```
In [6]: 1 fig, ax1 = plt.subplots(1, 1, figsize = (20, 2))
2 title_dcf['cell_type'].value_counts().plot.barh()
```

```
Out[6]: <matplotlib.axes._subplots.AxesSubplot at 0x2f1c1a5c50>
```



The new dataset includes 6,705 melanocytic nevus images, and 514 basal cell carcinoma images. In order to test to include more basal cell carcinoma images.

## Image Data Preprocessing

The original dataset file size was in excess of 200 with images sized at 650 x 450 x 300dpi. By closely examining the dataset, there are several advantages of resizing the images:

- Some of the images contain camera lens boundaries (which are black in the image), which will decrease the testing accuracy.
- The large file size requires a higher level of processing time.

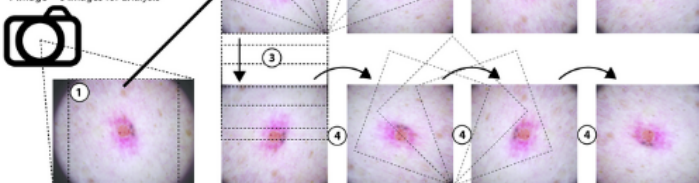
The new file sizes are standardized and cropped to 90 x 90 x 300 dpi.

In addition, we are able to increase the quantity of basal cell carcinoma files from the original 514 images. With the process of rotating (90, 180, and 270 degrees) and mirroring these images (rotating a second time), we are able to test a total of 4,112 images, which will provide a higher testing accuracy.

## Image preprocessing, and creating more image data for analysis:

1. Resampling and cropping original image
2. Rotating images
3. Mirroring images
4. Rotating again

1 image = 8 images for analysis



Separate data to targeted values (moles and cancerous skin cells)

New statistics

Image preprocessing, increasing dataset

## The Dataset used for Machine Learning

The final dataset that will be used contains the following information:

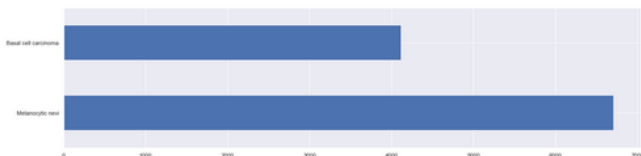
```
In [7]: 1 #Metadata information
2 title_dcf.describe(exclude=[np.number])
```

```
Out[7]:
```

	lesion_id	image_id	dx	dx_type	sex	localization	orientation	path	cell_type
count	10817	10817	10817	10817	10817	10817	0	10817	10817
unique	5720	10817	2	3	3	15	0	0	2
top	HAVI_0004821	nv3125	nv	hair	male	back	original	hav	Melanocytic nevus
freq	22	1	8705	3704	3726	1613	hav	6705	

```
In [8]: 1 fig, ax1 = plt.subplots(1, 1, figsize = (20, 2))
2 title_dcf['cell_type'].value_counts().plot.barh()
```

```
Out[8]: <matplotlib.axes._subplots.AxesSubplot at 0x2f1c04733580>
```



This proportion of images for testing accuracy is much better.

```
In [9]: 1 #Get check, remove for final
2 title_dcf['path']
```

```
Out[9]: 0      None
1      None
2      None
3      None
4      None
...
10812  None
10813  None
10814  None
10815  None
10816  None
Name: path, Length: 10817, dtype: object
```

Final data values used for analysis

## Choosing the machine learning platform and tuning the Convolutional Neural Network

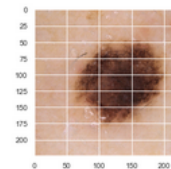
This process includes testing different layers and features from the prescribed CNN. In this project, we are using TensorFlow and Keras, and assigning various layers in Keras to determine which model can most accurately predict the differences between these two skin cells.

Now, the images will be loaded into memory for analysis

```
In [1]: 1 #Import libraries and testing with TensorFlow
2 import tensorflow as tf # Import tensorflow (Keras included)
3 import os
4 import cv2

In [2]: 1 #Show color and resampled image in grayscale
2 img = cv2.imread('C:/Users/admin/01_Business_Apps_of_HG(XS22)/skin_analysis/images/nv0008.jpg')
3 plt.imshow(cv2.cvtColor(img, cv2.COLOR_BGR2RGB))
4 plt.show()

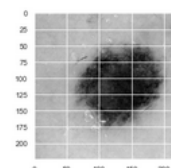
In [3]: 1 img = cv2.imread('C:/Users/admin/01_Business_Apps_of_HG(XS22)/skin_analysis/images/nv0008.jpg', cv2.IMREAD_GRAYSCALE)
2 imshow = cv2.imshow('imshow', img)
3 plt.imshow(new_array, cmap='gray')
```



Implementing the CNN using TensorFlow and Keras

Image preview, filter on secondary image

```
Out[3]: <matplotlib.image.AxesImage at 0x1e9a0c9e80>
```



```
In [6]: 1 #Create training data from images, this will take some time depending on size of data
2 training_data = []

3
4 def create_training_data():
5     for category in ['bas', 'nv']:
6         path = os.path.join('C:/Users/admin/01_Business_Apps_of_HG(XS22)/skin_analysis/images', category)
7         class_label = ['bas', 'nv']
8         for img in os.listdir(path):
9             try:
10                 img_array = cv2.imread(os.path.join(path, img), cv2.IMREAD_GRAYSCALE)
11                 new_array = cv2.resize(img_array, (256, 256))
12                 training_data.append((new_array, class_label))
13             except Exception as e:
14                 pass
15
16 create_training_data()
```

```
In [7]: 1 #Show total quantity of data/images that will be used for analysis
2 #Shuffle the images randomly for the training data
3 print(len(training_data))
4
5 import random
6
7 random.shuffle(training_data)
```

10817

Creating random training data from dataset

# Addendum B-Jupyter notebook excerpts (3)

```
In [8]: 1 #Show feature set for data
2 X = []
3 y = []
4
5 for features, label in training_data:
6     X.append(features)
7     y.append(label)
8
9 X = np.array(X).reshape(-1, 28*28,1)
10 y = np.array(y)
11
12 print(X.shape, y.shape)
13
14 (10817, 224, 224, 1) (10817,)
```

Show feature set for data

```
In [9]: 1 #Numpy core array assigned to the dataset for training and test datasets
2 import pickle
3 pickle_out = open('X.pickle', 'wb')
4 pickle.dump(X, pickle_out)
5 pickle_out.close()
6
7 pickle_out = open('y.pickle', 'wb')
8 pickle.dump(y, pickle_out)
9 pickle_out.close()
```

Implementing array assigning training and test datasets

```
In [10]: 1 #Tensorflow and Keras Convolutional Neural Network with imported layers for image detection
2 #Refer to Keras layer information at https://keras.io/api/models/sequential/#sequential-class
3 #Refer to Keras layer information at https://keras.io/api/layers/core_layers/
4 from tensorflow.keras.models import Sequential
5 from tensorflow.keras.layers import Activation, Flatten, Conv2D, Dense, Dropout, Flatten, MaxPooling2D
6 import pickle
```

Testing model performance on various hyperparameters

```
In [8]: 1 #Running CNN on assigned layers, training data= no test
2 tf.random.set_seed(0)
3
4 # define a sequential neural network
5 model = Sequential()
6
7 model.add(Flatten()) # Flatten image
8 model.add(Dense(1000, activation='relu')) # Adds a densely-connected layer with 1000 units
9 #model.add(Dense(100, activation='relu')) # Decided for this test
10 model.add(Dense(10, activation='softmax')) # Number of neurons final layer
11
12 model.compile(optimizer='adam', #adam #adamax,
13               loss='sparse_categorical_crossentropy',
14               metrics=['accuracy'])
15
16 model.fit(X,y, epochs=1)
17
18 Train on 10817 samples
19 10817/10817 [=====] - 85s 6ms/sample - loss: 2.7870415 - accuracy: 0.8545
20
21 Out[8]: <tensorflow.python.keras.callbacks.History at 0x241800286a0>
```

```
In [10]: 1 model.summary()
2
3 Model: "sequential"
4
5 Layer (type) Output Shape Param #
6 -----
7 Flatten (Flatten) multiple 0
8 Dense (Dense) multiple 50828000
9 Dense_1 (Dense) multiple 10010
10 -----
11 Total params: 50,838,010
12 Trainable params: 50,838,010
13 Non-trainable params: 0
```

```
In [40]: 1 #Modifying layers to test accuracy on different assigned layers
2 #Add validation parameter, currently assigned to 10% of images
3 x = pickle.load(open('X.pickle', 'rb'))
4 y = pickle.load(open('y.pickle', 'rb'))
5
6 X = x/225.0
7
8 model = Sequential()
9 model.add(Conv2D(64, (3,3), input_shape=(224,224,1)))
10
11 model.add(Activation('relu')) #ReLU - rectified linear unit, computationally efficient
12 model.add(MaxPooling2D(pool_size=(2,2)))
13 model.add(Flatten())
14 model.add(Dense(64))
15 model.add(Dense(1))
16 model.add(Activation('sigmoid'))
17
18 model.compile(loss='binary_crossentropy',
19               optimizer='adam',
20               metrics=['accuracy'])
21
22 history = model.fit(X,y, epochs=2, validation_split=0.1)
```

Testing model performance on various hyperparameters

```
Train on 9735 samples, validate on 1082 samples
Epoch 1/2
9735/9735 [=====] - 342s 30ms/sample - loss: 3.7100 - accuracy: 0.7405 - val_loss: 0.5221 - val_accuracy: 0.7532
Epoch 2/2
9735/9735 [=====] - 357s 40ms/sample - loss: 0.6104 - accuracy: 0.8150 - val_loss: 0.4033 - val_accuracy: 0.8057
```

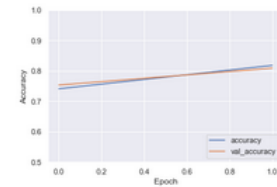
```
In [42]: 1 model.summary()
```

Layer (type)	Output Shape	Param #
conv2d_3 (Conv2D)	(None, 224, 224, 64)	640
activation_6 (Activation)	(None, 224, 224, 64)	0
max_pooling2d_3 (MaxPooling2D)	(None, 112, 112, 64)	0
flatten_8 (Flatten)	(None, 785744)	0
dense_10 (Dense)	(None, 64)	50466560
dense_11 (Dense)	(None, 1)	65
activation_7 (Activation)	(None, 1)	0
<b>Total params:</b>	<b>50,487,265</b>	
<b>Trainable params:</b>	<b>50,487,265</b>	
<b>Non-trainable params:</b>	<b>0</b>	

Reviewing performance and accuracy results

```
In [41]: 1 plt.plot(history.history['accuracy'], label='accuracy')
2 plt.plot(history.history['val_accuracy'], label='val_accuracy')
3 plt.xlabel('Epoch')
4 plt.ylabel('Accuracy')
5 plt.ylim([0.5, 1])
6 plt.legend(loc='lower right')
```

```
Out[41]: <matplotlib.legend.Legend at 0x2f4d0b8d0>
```



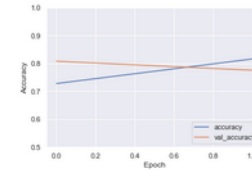
```
In [18]: 1 #Modifying layers to test accuracy on different assigned layers
2 #Add validation parameter, currently assigned to 20% of images
3 x = pickle.load(open('X.pickle', 'rb'))
4 y = pickle.load(open('y.pickle', 'rb'))
5
6 X = x/225.0
7
8 model = Sequential()
9 model.add(Conv2D(64, (3,3), input_shape=(224,224,1)))
10
11 model.add(Activation('relu')) #ReLU - rectified linear unit, computationally efficient
12 model.add(MaxPooling2D(pool_size=(2,2)))
13 model.add(Flatten())
14 model.add(Dense(64))
15 model.add(Dense(1))
16 model.add(Activation('sigmoid'))
17
18 model.compile(loss='binary_crossentropy',
19               optimizer='adam',
20               metrics=['accuracy'])
21
22 history = model.fit(X,y, epochs=2, validation_split=0.2)
```

Second test, varying hyperparameters

```
Train on 9553 samples, validate on 2164 samples
Epoch 1/2
9553/9553 [=====] - 268s 31ms/sample - loss: 5.4297 - accuracy: 0.7276 - val_loss: 0.4411 - val_accuracy: 0.8076
Epoch 2/2
9553/9553 [=====] - 268s 31ms/sample - loss: 0.4069 - accuracy: 0.8159 - val_loss: 0.4491 - val_accuracy: 0.7754
```

```
In [22]: 1 plt.plot(history.history['accuracy'], label='accuracy')
2 plt.plot(history.history['val_accuracy'], label='val_accuracy')
3 plt.xlabel('Epoch')
4 plt.ylabel('Accuracy')
5 plt.ylim([0.5, 1])
6 plt.legend(loc='lower right')
```

```
Out[22]: <matplotlib.legend.Legend at 0x231e7e48320>
```



Reviewing performance and accuracy results

```
In [11]: 1 #Modifying layers to test accuracy on different assigned layers
2 #Add validation parameter, currently assigned to 20% of images
3 x = pickle.load(open('X.pickle', 'rb'))
4 y = pickle.load(open('y.pickle', 'rb'))
5
6 X = x/225.0
7
8 model = Sequential()
9 model.add(Conv2D(64, (3,3), input_shape=(224,224,1)))
10
11 model.add(Activation('relu')) #ReLU - rectified linear unit, computationally efficient
12 model.add(MaxPooling2D(pool_size=(2,2)))
13 model.add(Flatten())
14 model.add(Dense(1000))
15 #model.add(Dense(32))
16 model.add(Dense(1))
17 model.add(Activation('sigmoid'))
18
19 model.compile(loss='binary_crossentropy',
20               optimizer='adam',
21               metrics=['accuracy'])
22
23 history = model.fit(X,y, epochs=2, validation_split=0.2)
24
25 Train on 9553 samples, validate on 2164 samples
Epoch 1/2
9553/9553 [=====] - 1324s 16ms/sample - loss: 6.7587 - accuracy: 0.7474 - val_loss: 0.4534 - val_accuracy: 0.7935
Epoch 2/2
9553/9553 [=====] - 1344s 15ms/sample - loss: 0.3990 - accuracy: 0.8245 - val_loss: 0.4599 - val_accuracy: 0.8116
```

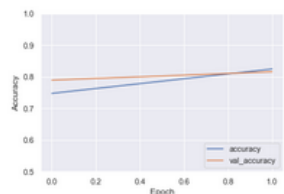
Third test, varying hyperparameters



# Addendum B-Jupyter notebook excerpts (4)

```
In [12]: 1 plt.plot(history.history['accuracy'], label='accuracy')
2 plt.plot(history.history['val_accuracy'], label = 'val_accuracy')
3 plt.xlabel('Epoch')
4 plt.ylabel('Accuracy')
5 plt.ylim([0.5, 1])
6 plt.legend(loc='lower right')
```

Out[12]: <matplotlib.legend.Legend at 0x2418d520fe>



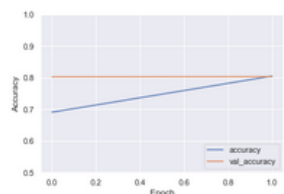
Reviewing  
performance  
and accuracy  
results

```
In [20]: 1 #Modifying layers to test accuracy on different assigned layers
2 #Add validation parameter, currently assigned to 20% of images
3 x = pickle.load(open('X.pickle', 'rb'))
4 y = pickle.load(open('y.pickle', 'rb'))
5
6 X = x/225.0
7
8 model = Sequential()
9 model.add(Conv2D(64, (3,3), input_shape=(225,225,1)))
10
11 model.add(Activation('relu'))
12 model.add(MaxPooling2D(pool_size=(2,2)))
13 model.add(Flatten())
14 model.add(Dense(1000))
15 model.add(Dense(500))
16 model.add(Dense(1))
17 model.add(Activation('sigmoid'))
18
19
20 model.compile(loss='binary_crossentropy',
21               optimizer='adam',
22               metrics=['accuracy'])
23
24 history = model.fit(X,y, epochs=2, validation_split=0.2)
```

Train on 5653 samples, validate on 1164 samples  
Epoch 1/2  
5653/5653 [=====] - 1351s 158ms/sample - loss: 11.3755 - accuracy: 0.6905 - val\_loss: 0.4543 - val\_accuracy: 0.8027  
Epoch 2/2  
5653/5653 [=====] - 1331s 146ms/sample - loss: 0.4402 - accuracy: 0.8048 - val\_loss: 0.4364 - val\_accuracy: 0.8031

```
In [21]: 1 plt.plot(history.history['accuracy'], label='accuracy')
2 plt.plot(history.history['val_accuracy'], label = 'val_accuracy')
3 plt.xlabel('Epoch')
4 plt.ylabel('Accuracy')
5 plt.ylim([0.5, 1])
6 plt.legend(loc='lower right')
```

Out[21]: <matplotlib.legend.Legend at 0x2418d50586>



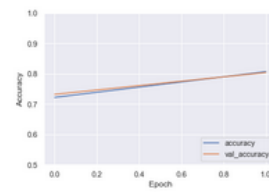
Fourth test,  
varying  
hyperparameter  
rs

```
In [11]: 1 #Final Model test, chose model 2, now running 10 epochs
2 #Modifying layers to test accuracy on different assigned layers
3 #Add validation parameter, currently assigned to 20% of images
4 x = pickle.load(open('X.pickle', 'rb'))
5 y = pickle.load(open('y.pickle', 'rb'))
6
7 X = x/225.0
8
9 model = Sequential()
10 model.add(Conv2D(64, (3,3), input_shape=(225,225,1)))
11
12 model.add(Activation('relu'))
13 model.add(MaxPooling2D(pool_size=(2,2)))
14 model.add(Flatten())
15 model.add(Dense(1000))
16 model.add(Dense(500))
17 model.add(Dense(1))
18 model.add(Activation('sigmoid'))
19
20 model.compile(loss='binary_crossentropy',
21               optimizer='adam',
22               metrics=['accuracy'])
23
24 history = model.fit(X,y, epochs=2, validation_split=0.2)
```

Train on 5653 samples, validate on 1164 samples  
Epoch 1/2  
5653/5653 [=====] - 137s 58ms/sample - loss: 7.5275 - accuracy: 0.7219 - val\_loss: 0.5086 - val\_accuracy: 0.7320  
Epoch 2/2  
5653/5653 [=====] - 722s 54ms/sample - loss: 0.4463 - accuracy: 0.8069 - val\_loss: 0.4266 - val\_accuracy: 0.8041

```
In [12]: 1 plt.plot(history.history['accuracy'], label='accuracy')
2 plt.plot(history.history['val_accuracy'], label = 'val_accuracy')
3 plt.xlabel('Epoch')
4 plt.ylabel('Accuracy')
5 plt.ylim([0.5, 1])
6 plt.legend(loc='lower right')
```

Out[12]: <matplotlib.legend.Legend at 0x1a5ef07d30>



Fifth test,  
varying  
hyperparameter  
rs

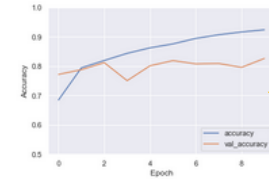
Reviewing  
performance  
and accuracy  
results

```
In [13]: 1 #Modifying layers to test accuracy on different assigned layers
2 #Add validation parameter, currently assigned to 20% of images
3 x = pickle.load(open('X.pickle', 'rb'))
4 y = pickle.load(open('y.pickle', 'rb'))
5
6 X = x/225.0
7
8 model = Sequential()
9 model.add(Conv2D(64, (3,3), input_shape=(225,225,1)))
10
11 model.add(Activation('relu'))
12 model.add(MaxPooling2D(pool_size=(2,2)))
13 model.add(Flatten())
14 model.add(Dense(1000))
15 model.add(Dense(500))
16 model.add(Dense(1))
17 model.add(Activation('sigmoid'))
18
19 model.compile(loss='binary_crossentropy',
20               optimizer='adam',
21               metrics=['accuracy'])
22
23 history = model.fit(X,y, epochs=10, validation_split=0.2)
```

Train on 5653 samples, validate on 1164 samples  
Epoch 1/10  
5653/5653 [=====] - 1353s 158ms/sample - loss: 22.9174 - accuracy: 0.6545 - val\_loss: 0.4565 - val\_accuracy: 0.7717  
Epoch 2/10  
5653/5653 [=====] - 1329s 154ms/sample - loss: 0.4803 - accuracy: 0.7943 - val\_loss: 0.4802 - val\_accuracy: 0.7979  
Epoch 3/10  
5653/5653 [=====] - 1343s 153ms/sample - loss: 0.4104 - accuracy: 0.8193 - val\_loss: 0.4831 - val\_accuracy: 0.8119  
Epoch 4/10  
5653/5653 [=====] - 1352s 158ms/sample - loss: 0.3808 - accuracy: 0.8436 - val\_loss: 0.5051 - val\_accuracy: 0.7935  
Epoch 5/10  
5653/5653 [=====] - 1300s 150ms/sample - loss: 0.3224 - accuracy: 0.8624 - val\_loss: 0.4634 - val\_accuracy: 0.8033  
Epoch 6/10  
5653/5653 [=====] - 1316s 153ms/sample - loss: 0.2948 - accuracy: 0.8758 - val\_loss: 0.4708 - val\_accuracy: 0.8139  
Epoch 7/10  
5653/5653 [=====] - 1335s 154ms/sample - loss: 0.2599 - accuracy: 0.8945 - val\_loss: 0.5123 - val\_accuracy: 0.8078  
Epoch 8/10  
5653/5653 [=====] - 1355s 157ms/sample - loss: 0.2305 - accuracy: 0.9070 - val\_loss: 0.4801 - val\_accuracy: 0.8081  
Epoch 9/10  
5653/5653 [=====] - 1395s 161ms/sample - loss: 0.2115 - accuracy: 0.9164 - val\_loss: 0.5544 - val\_accuracy: 0.7957  
Epoch 10/10  
5653/5653 [=====] - 1371s 159ms/sample - loss: 0.2015 - accuracy: 0.9240 - val\_loss: 0.5419 - val\_accuracy: 0.8282

```
In [14]: 1 plt.plot(history.history['accuracy'], label='accuracy')
2 plt.plot(history.history['val_accuracy'], label = 'val_accuracy')
3 plt.xlabel('Epoch')
4 plt.ylabel('Accuracy')
5 plt.ylim([0.5, 1])
6 plt.legend(loc='lower right')
```

Out[14]: <matplotlib.legend.Legend at 0x1a5ef07d30>



Reviewing performance  
and accuracy results. This  
model had a 92.4%  
accuracy rating for  
predicting the difference  
between a normal and  
cancerous skin cell.  
(slightly better than  
initial 91.25%)



# Thank you

**Keywords:** applications, algorithms, analytics, convolutional neural network (CNN), data science, databases, data processing, dataset, deep learning, dermatology, digital health, health information, image processing, machine learning, model tuning, python, skin cancer, skin cells, skin conditions, tensorflow.

# Attribution-NonCommercial-ShareAlike 4.0 International



Creative Commons Corporation ("Creative Commons") is not a law firm and does not provide legal services or legal advice. Distribution of Creative Commons public licenses does not create a lawyer-client or other relationship. Creative Commons makes its licenses and related information available on an "as-is" basis. Creative Commons gives no warranties regarding its licenses, any material licensed under their terms and conditions, or any related information. Creative Commons disclaims all liability for damages resulting from their use to the fullest extent possible.

## Using Creative Commons Public Licenses

Creative Commons public licenses provide a standard set of terms and conditions that creators and other rights holders may use to share original works of authorship and other material subject to copyright and certain other rights specified in the public license below. The following considerations are for informational purposes only, are not exhaustive, and do not form part of our licenses.

Considerations for licensors: Our public licenses are intended for use by those authorized to give the public permission to use material in ways otherwise restricted by copyright and certain other rights. Our licenses are irrevocable. Licensors should read and understand the terms and conditions of the license they choose before applying it. Licensors should also secure all rights necessary before applying our licenses so that the public can reuse the material as expected. Licensors should clearly mark any material not subject to the license. This includes other CC-licensed material, or material used under an exception or limitation to copyright. More considerations for licensors:

[wiki.creativecommons.org/Considerations\\_for\\_licensors](http://wiki.creativecommons.org/Considerations_for_licensors)

Considerations for the public: By using one of our public licenses, a licensor grants the public permission to use the licensed material under specified terms and conditions. If the licensor's permission is not necessary for any reason—for example, because of any applicable exception or limitation to copyright--then that use is not regulated by the license. Our licenses grant only permissions under copyright and certain other rights that a licensor has authority to grant. Use of the licensed material may still be restricted for other reasons, including because others have copyright or other rights in the material. A licensor may make special requests, such as asking that all changes be marked or described. Although not required by our licenses, you are encouraged to respect those requests where reasonable. More considerations for the public:

[wiki.creativecommons.org/Considerations\\_for\\_licensees](http://wiki.creativecommons.org/Considerations_for_licensees)

## Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International Public License

By exercising the Licensed Rights (defined below), You accept and agree to be bound by the terms and conditions of this Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International Public License ("Public License"). To the extent this Public License may be interpreted as a contract, You are granted the Licensed Rights in consideration of Your acceptance of these terms and conditions, and the Licensor grants You such rights in consideration of benefits the Licensor receives from making the Licensed Material available under these terms and conditions.

### Section 1 -- Definitions.

a. Adapted Material means material subject to Copyright and Similar Rights that is derived from or based upon the Licensed Material and in which the Licensed Material is translated, altered, arranged, transformed, or otherwise modified in a manner requiring permission under the Copyright and Similar Rights held by the Licensor. For purposes of this Public License, where the Licensed Material is a musical work, performance, or sound recording, Adapted Material is always produced where the Licensed Material is synced in timed relation with a moving image.

# Attribution-NonCommercial-ShareAlike 4.0 International



- b. Adapter's License means the license You apply to Your Copyright and Similar Rights in Your contributions to Adapted Material in accordance with the terms and conditions of this Public License.
- c. BY-NC-SA Compatible License means a license listed at [creativecommons.org/compatiblelicenses](https://creativecommons.org/compatiblelicenses), approved by Creative Commons as essentially the equivalent of this Public License.
- d. Copyright and Similar Rights means copyright and/or similar rights closely related to copyright including, without limitation, performance, broadcast, sound recording, and Sui Generis Database Rights, without regard to how the rights are labeled or categorized. For purposes of this Public License, the rights specified in Section 2(b)(1)-(2) are not Copyright and Similar Rights.
- e. Effective Technological Measures means those measures that, in the absence of proper authority, may not be circumvented under laws fulfilling obligations under Article 11 of the WIPO Copyright Treaty adopted on December 20, 1996, and/or similar international agreements.
- f. Exceptions and Limitations means fair use, fair dealing, and/or any other exception or limitation to Copyright and Similar Rights that applies to Your use of the Licensed Material.
- g. License Elements means the license attributes listed in the name of a Creative Commons Public License. The License Elements of this Public License are Attribution, NonCommercial, and ShareAlike.
- h. Licensed Material means the artistic or literary work, database, other material to which the Licenser applied this Public License.
- i. Licensed Rights means the rights granted to You subject to the terms and conditions of this Public License, which are limited to all Copyright and Similar Rights that apply to Your use of the Licensed Material and that the Licenser has authority to license.
- j. Licenser means the individual(s) or entity(ies) granting rights under this Public License.
- k. NonCommercial means not primarily intended for or directed towards commercial advantage or monetary compensation. For purposes of this Public License, the exchange of the Licensed Material for other material subject to Copyright and Similar Rights by digital file-sharing or similar means is NonCommercial provided there is no payment of monetary compensation in connection with the exchange.
- l. Share means to provide material to the public by any means or process that requires permission under the Licensed Rights, such as reproduction, public display, public performance, distribution, dissemination, communication, or importation, and to make material available to the public including in ways that members of the public may access the material from a place and at a time individually chosen by them.
- m. Sui Generis Database Rights means rights other than copyright resulting from Directive 96/9/EC of the European Parliament and of the Council of 11 March 1996 on the legal protection of databases, as amended and/or succeeded, as well as other essentially equivalent rights anywhere in the world.
- n. You means the individual or entity exercising the Licensed Rights under this Public License. Your has a corresponding meaning.

## Section 2 -- Scope.

- a. License grant.  
Subject to the terms and conditions of this Public License, Licenser hereby grants You a worldwide, royalty-free, non-sublicensable, non-exclusive, irrevocable license to exercise the Licensed Rights in the Licensed Material to:
  - a. reproduce and Share the Licensed Material, in whole or in part, for NonCommercial purposes only; and
  - b. produce, reproduce, and Share Adapted Material for NonCommercial purposes only.

# Attribution-NonCommercial-ShareAlike 4.0 International



2. Exceptions and Limitations. For the avoidance of doubt, where Exceptions and Limitations apply to Your use, this Public License does not apply, and You do not need to comply with its terms and conditions.

3. Term. The term of this Public License is specified in Section 6(a).

4. Media and formats; technical modifications allowed. The Licensor authorizes You to exercise the Licensed Rights in all media and formats whether now known or hereafter created, to make technical modifications necessary to do so. The Licensor waives and/or agrees not to assert any right or authority to forbid You from making technical modifications necessary to exercise the Licensed Rights, including technical modifications necessary to circumvent Effective Technological Measures. For purposes of this Public License, making modifications authorized by this Section 2(a) (4) never produces Adapted Material.

5. Downstream recipients.

a. Offer from the Licensor -- Licensed Material. Every recipient of the Licensed Material automatically receives an offer from the Licensor to exercise the Licensed Rights under the terms and conditions of this Public License.

b. Additional offer from the Licensor -- Adapted Material. Every recipient of Adapted Material from You automatically receives an offer from the Licensor to exercise the Licensed Rights in the Adapted Material under the conditions of the Adapter's License You apply.

c. No downstream restrictions. You may not offer or impose any additional or different terms or conditions on, or apply any Effective Technological Measures to, the Licensed Material if doing so restricts exercise of the Licensed Rights by any recipient of the Licensed Material.

6. No endorsement. Nothing in this Public License constitutes or may be construed as permission to assert or imply that You are, or that Your use of the Licensed Material is, connected with, or sponsored, endorsed, or granted official status by, Licensor or others designated to receive attribution as provided in Section 3(a)(1)(A)(i).

b. Other rights.

Moral rights, such as the right of integrity, are not licensed under this Public License, nor are publicity, privacy, and/or other similar personality rights; however, to the extent possible, the Licensor waives and/or agrees not to assert any such rights held by the Licensor to the limited extent necessary to allow You to exercise the Licensed Rights, but not otherwise.

2. Patent and trademark rights are not licensed under this Public License.

3. To the extent possible, the Licensor waives any right to collect royalties from You for the exercise of the Licensed Rights, whether directly or through a collecting society under any voluntary or waivable statutory or compulsory licensing scheme. In all other cases the Licensor expressly reserves any right to collect such royalties, including when the Licensed Material is used other than for NonCommercial purposes.

## Section 3 -- License Conditions.

Your exercise of the Licensed Rights is expressly made subject to the following conditions.

a. Attribution.

If You Share the Licensed Material (including in modified form), You must:

a. retain the following if it is supplied by the Licensor with the Licensed Material:

# Attribution-NonCommercial-ShareAlike 4.0 International



- i. identification of the creator(s) of the Licensed Material and any others designated to receive attribution, in any reasonable manner requested by the Licensor (including by pseudonym if designated);
    - ii. a copyright notice;
    - iii. a notice that refers to this Public License;
    - iv. a notice that refers to the disclaimer of warranties;
    - v. a URI or hyperlink to the Licensed Material to the extent reasonably practicable;
  - b. indicate if You modified the Licensed Material and retain an indication of any previous modifications; and
  - c. indicate the Licensed Material is licensed under this Public License, and include the text of, or the URI or hyperlink to, this Public License.
2. You may satisfy the conditions in Section 3(a)(1) in any reasonable manner based on the medium, means, and context in which You Share the Licensed Material. For example, it may be reasonable to satisfy the conditions by providing a URI or hyperlink to a resource that includes the required information.
3. If requested by the Licensor, You must remove any of the information required by Section 3(a)(1)(A) to the extent reasonably practicable.
- b. ShareAlike.

In addition to the conditions in Section 3(a), if You Share Adapted Material You produce, the following conditions also apply.

The Adapter's License You apply must be a Creative Commons license with the same License Elements, this version or later, or a BY-NC-SA Compatible License.
  2. You must include the text of, or the URI or hyperlink to, the Adapter's License You apply. You may satisfy this condition in any reasonable manner based on the medium, means, and context in which You Share Adapted Material.
  3. You may not offer or impose any additional or different terms conditions on, or apply any Effective Technological Measures to, Adapted Material that restrict exercise of the rights granted under the Adapter's License You apply.

## Section 4 -- Sui Generis Database Rights.

Where the Licensed Rights include Sui Generis Database Rights that apply to Your use of the Licensed Material:

- a. for the avoidance of doubt, Section 2(a)(1) grants You the right to extract, reuse, reproduce, and Share all or a substantial portion of the contents of the database for NonCommercial purposes only;
- b. if You include all or a substantial portion of the database contents in a database in which You have Sui Generis Database Rights, then the database in which You have Sui Generis Database Rights (but not its individual contents) is Adapted Material, including for purposes of Section 3(b); and
- c. You must comply with the conditions in Section 3(a) if You Share or a substantial portion of the contents of the database.

For the avoidance of doubt, this Section 4 supplements and does not replace Your obligations under this Public License where the Licensed Rights include other Copyright and Similar Rights.



# Attribution-NonCommercial-ShareAlike 4.0 International



## Section 5 -- Disclaimer of Warranties and Limitation of Liability.

- a. UNLESS OTHERWISE SEPARATELY UNDERTAKEN BY THE LICENSOR, TO THE EXTENT POSSIBLE, THE LICENSOR OFFERS THE LICENSED MATERIAL AS-IS AND AS-AVAILABLE, AND MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND CONCERNING THE LICENSED MATERIAL, WHETHER EXPRESS, IMPLIED, STATUTORY, OR OTHER. THIS INCLUDES, WITHOUT LIMITATION, WARRANTIES OF TITLE, MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NON-INFRINGEMENT, ABSENCE OF LATENT OR OTHER DEFECTS, ACCURACY, OR THE PRESENCE OR ABSENCE OF ERRORS, WHETHER OR NOT KNOWN OR DISCOVERABLE. WHERE DISCLAIMERS OF WARRANTIES ARE NOT ALLOWED IN FULL OR IN PART, THIS DISCLAIMER MAY NOT APPLY TO YOU.
- b. TO THE EXTENT POSSIBLE, IN NO EVENT WILL THE LICENSOR BE LIABLE TO YOU ON ANY LEGAL THEORY (INCLUDING, WITHOUT LIMITATION, NEGLIGENCE) OR OTHERWISE FOR ANY DIRECT, SPECIAL, INDIRECT, INCIDENTAL, CONSEQUENTIAL, PUNITIVE, EXEMPLARY, OR OTHER LOSSES, COSTS, EXPENSES, OR DAMAGES ARISING OUT OF THIS PUBLIC LICENSE OR USE OF THE LICENSED MATERIAL, EVEN IF THE LICENSOR HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH LOSSES, COSTS, EXPENSES, OR DAMAGES. WHERE A LIMITATION OF LIABILITY IS NOT ALLOWED IN FULL OR IN PART, THIS LIMITATION MAY NOT APPLY TO YOU.
- c. The disclaimer of warranties and limitation of liability provided above shall be interpreted in a manner that, to the extent possible, most closely approximates an absolute disclaimer and waiver of all liability.

## Section 6 -- Term and Termination.

- a. This Public License applies for the term of the Copyright and Similar Rights licensed here. However, if You fail to comply with this Public License, then Your rights under this Public License terminate automatically.
- b. Where Your right to use the Licensed Material has terminated under Section 6(a), it reinstates:
  1. automatically as of the date the violation is cured, provided it is cured within 30 days of Your discovery of the violation; or
  2. upon express reinstatement by the Licensor.For the avoidance of doubt, this Section 6(b) does not affect any right the Licensor may have to seek remedies for Your violations of this Public License.
- c. For the avoidance of doubt, the Licensor may also offer the Licensed Material under separate terms or conditions or stop distributing the Licensed Material at any time; however, doing so will not terminate this Public License.
- d. Sections 1, 5, 6, 7, and 8 survive termination of this Public License.

# Attribution-NonCommercial-ShareAlike 4.0 International



## Section 7 -- Other Terms and Conditions.

- a. The Licensor shall not be bound by any additional or different terms or conditions communicated by You unless expressly agreed.
- b. Any arrangements, understandings, or agreements regarding the Licensed Material not stated herein are separate from and independent of the terms and conditions of this Public License.

## Section 8 -- Interpretation.

- a. For the avoidance of doubt, this Public License does not, and shall not be interpreted to, reduce, limit, restrict, or impose conditions on any use of the Licensed Material that could lawfully be made without permission under this Public License.
- b. To the extent possible, if any provision of this Public License is deemed unenforceable, it shall be automatically reformed to the minimum extent necessary to make it enforceable. If the provision cannot be reformed, it shall be severed from this Public License without affecting the enforceability of the remaining terms and conditions.
- c. No term or condition of this Public License will be waived and no failure to comply consented to unless expressly agreed to by the Licensor.
- d. Nothing in this Public License constitutes or may be interpreted as a limitation upon, or waiver of, any privileges and immunities that apply to the Licensor or You, including from the legal processes of any jurisdiction or authority.

=====

Creative Commons is not a party to its public licenses. Notwithstanding, Creative Commons may elect to apply one of its public licenses to material it publishes and in those instances will be considered the "Licensor." The text of the Creative Commons public licenses is dedicated to the public domain under the CC0 Public Domain Dedication. Except for the limited purpose of indicating that material is shared under a Creative Commons public license or as otherwise permitted by the Creative Commons policies published at [creativecommons.org/policies](https://creativecommons.org/policies), Creative Commons does not authorize the use of the trademark "Creative Commons" or any other trademark or logo of Creative Commons without its prior written consent including, without limitation, in connection with any unauthorized modifications to any of its public licenses or any other arrangements, understandings, or agreements concerning use of licensed material. For the avoidance of doubt, this paragraph does not form part of the public licenses.

This data may also be protected by the California Consumer Privacy Act (CCPA) and other legal restriction depending upon your jurisdiction.

The data is publicly available and shared within the works cited section. Please download the data for yourself and create meaningful value, while respecting the investment of countless hours, training, education, time, and processes used to develop this publication as a digital product. Thank you for taking time to read, learn, and hopefully inspiration to contribute towards a better future for all of us.

