Classifying Skin Lesions with Artificial Neural Networks

Juliana Negrini de Araujo

Summary

- Introduction and Literature Review
- Dataset
- Models
- Results

Skin cancer is currently the most common type of cancer. For the UK, over a hundred thousand patients are diagnosed with some type of skin cancer each year;

Melanoma skin cancer is the 20th most common cause of cancer death, being accountable for 2.400 deaths every year;

First well known work for skin lesion detection was published by Stanford University, achieving 66% accuracy. Currently, top performing model is 89%;

Three different NN models on the HAM10000 dataset.

Introduction and Literature Review

Dataset

- Total 10.000 images
- Seven Classes
- Most dangerous types of lesion are Basal Cell Carcinoma and Melanoma
- Highly unbalanced dataset
- Two approaches to fix unbalance data

Class	Description	Number of samples	% of class samples
akiec	Actinic Keratoses (Solar Keratoses) and Intraepithelial Carcinoma (Bowen's disease) skin lesions. Usually caused by sun damage and treated without the need for surgery.	327	3.27%
bcc	Basal Cell Carcinoma is the most common type of skin cancer. Treatment is required, but it rarely metastasizes.	514	5.13%
bkl	Seborrheic Keratoses, Solar Lentigo and Lichen-planus-like Keratoses (LPLK) benign samples. They usually require biopsy due to similarities with melanoma lesions.	1099	10.97%
df	Dermatofibroma lesions.	115	1.15%
nv	Class contains images extensive data on variants of benign neoplasms called Melanocytic Nevi.	6705	66.95%
mel	Malignant Melanoma. Surgical removal in the early stage of cancer can provide a cure.	1113	11.11%
vasc	This class includes vascular skin lesions such as Cherry Angiomas and Angiokeratomas. It is also a benign type of skin lesion, only treated if it causes discomfort to the patient.	142	1.42%

Added an extra of 185 images of Melanoma (mel), Seborrheic Keratoses (bkl), Dermatofibroma (df) and Actinic Keratoses (akiec).

Created secondary dataset called HAM10000+

MLP

- Optimisers
- Dropout
- Batch Normalisation
- Layers and Neuros

Layer (type)	Output Shape	Param #
flatten_2 (Flatten)	(None, 150528)	0
batch_normalization_4 (Batch	(None, 150528)	602112
dense_4 (Dense)	(None, 128)	19267712
dropout_3 (Dropout)	(None, 128)	0
batch_normalization_5 (Batch	(None, 128)	512
dense_5 (Dense)	(None, 512)	66048
dropout_4 (Dropout)	(None, 512)	0
batch_normalization_6 (Batch	(None, 512)	2048
dense_6 (Dense)	(None, 7)	3591

Total params: 19,942,023
Trainable params: 19,639,687
Non-trainable params: 302,336

CNN

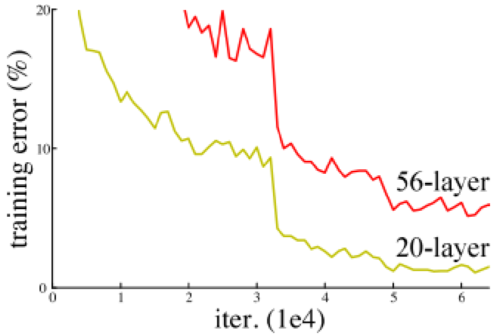
- 7 Convolutional Layers
- Dropout
- Batch Normalisation
- Average Pooling
- Reduce LR on Plateau
- Optimiser

Layer (type)	Output Shape	Param #
conv2d_36 (Conv2D)	(None, 75, 100, 32)	896
batch_normalization_46 (Batc	(None, 75, 100, 32)	128
conv2d_37 (Conv2D)	(None, 75, 100, 32)	9248
batch_normalization_47 (Batc	(None, 75, 100, 32)	128
average_pooling2d_21 (Averag	(None, 37, 50, 32)	0
dropout_26 (Dropout)	(None, 37, 50, 32)	0
conv2d_38 (Conv2D)	(None, 37, 50, 64)	18496
batch_normalization_48 (Batc	(None, 37, 50, 64)	256
conv2d_39 (Conv2D)	(None, 37, 50, 64)	36928
batch_normalization_49 (Batc	(None, 37, 50, 64)	256
average_pooling2d_22 (Averag	(None, 18, 25, 64)	0
dropout_27 (Dropout)	(None, 18, 25, 64)	0
conv2d_40 (Conv2D)	(None, 18, 25, 32)	18464
batch_normalization_50 (Batc	(None, 18, 25, 32)	128
average_pooling2d_23 (Averag	(None, 9, 12, 32)	0
dropout_28 (Dropout)	(None, 9, 12, 32)	0
conv2d_41 (Conv2D)	(None, 9, 12, 64)	18496
batch_normalization_51 (Batc	(None, 9, 12, 64)	256
conv2d_42 (Conv2D)	(None, 9, 12, 64)	36928
batch_normalization_52 (Batc	(None, 9, 12, 64)	256
average_pooling2d_24 (Averag	(None, 4, 6, 64)	0
dropout_29 (Dropout)	(None, 4, 6, 64)	0
flatten_6 (Flatten)	(None, 1536)	0
batch_normalization_53 (Batc	(None, 1536)	6144
dense_11 (Dense)	(None, 128)	196736
dropout_30 (Dropout)	(None, 128)	0
batch_normalization_54 (Batc	(None, 128)	512
dense_12 (Dense)	(None, 7)	903

Total params: 345,159
Trainable params: 341,127
Non-trainable params: 4,032

ResNet-50

- the ResNet tries to learn the residual function denoted by F(x) = H(x) – x. The hypothesis is that the residual function could be easier to learn than H(x).
- ILSVRC 2015 (ImageNet) and COCO 2015.
- reuse the weights of the pre-trained network
- Best results when the last 25 layers were re-trained with our dataset



7×7, 64, stride 2

3×3 max pool, stride 2

$$\left\{\begin{array}{c}1\times1, 64\\ 3\times3, 64\\ 1\times1, 256\end{array}\right\} \text{X } 3$$

$$\left\{\begin{array}{c}1\times1, 128\\ 3\times3, 128\\ 1\times1, 512\end{array}\right\} \text{X } 4$$

$$\left\{\begin{array}{c}1\times1, 256\\ 3\times3, 256\\ 1\times1, 1024\end{array}\right\} \text{X } 6$$

$$\left\{\begin{array}{c}1\times1, 512\\ 3\times3, 512\\ 1\times1, 2048\end{array}\right\} \text{X } 3$$

average pool, 1000-d fc, softmax

K. He, et al (2015)

Results

- Batch size of 10
- 50 epochs
- The final model results are defined as the average of three runs

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN}$$

$$Sensitivity = \frac{TP}{TP + FN}$$

$$Precision = \frac{TP}{TP + FP}$$

Same model as CNN but using additional data

Metric	Model			
	MLP	ResNet-50	CNN	CNN HAM10000 +
Accuracy	72.81%	73.37%	77.96%	78.68%
Sensitivity	39.67%	38.57%	50.57%	59.19%
Precision	54.52%	53.90%	56.57%	68.43%

Results

- Adding images was the most effective approach to improve accuracy

CNN

Class	Sensitivity	Precision
Actinic keratoses	37.33%	55.67%
Basal cell carcinoma	55.33%	66.67%
Benign keratosis-like	59.67%	54.00%
Dermatofibroma	6.67%	8.33%
Melanocytic nevi	95.00%	85.00%
Melanoma	21.00%	61.33%
Vascular Lesions	79.00%	65.00%

CNN HAM10000+

Class	Sensitivity	Precision
Actinic keratoses	51.33%	53.67%
Basal cell carcinoma	59.00%	76.33%
Benign keratosis-like	50.67%	59.67%
Dermatofibroma	52.33%	72.33%
Melanocytic nevi	95.00%	84.67%
Melanoma	32.67%	53.33%
Vascular Lesions	73.33%	79.00%

Results

- Sensitivity always rises at the cost of precision and vice-versa
- Most impressive improvement was the Dermatofibroma

Discussion and Conclusion

Single model accuracy seemed to achieve a plateau. Parameter fine-tune did not seem to effect the accuracy anymore

Usage of pre-trained ResNet-50 did not show major improvements

Adding only 185 of four classes was very effective

From the Literature Review its concluded that ensemble models and pre-trained architectures are the way forward

Future Work

Ensemble Models

Apply different pre-trained models than ResNet

Investigating new ways to balance image datasets, Focal Loss or application of GAN’s to generate new samples

From the Literature Review its concluded that ensemble models and pre-trained architectures are the way forward