JOURNAL OF ALGEBRAIC STATISTICS
Volume 13, No. 1, 2022, p. 511-516
https://publishoa.com
ISSN: 1309-3452

Trlu: A Customized Activation Function to Detect Erythemato-Squamous Skin Cancer at Early Stage

Rajashekar Deva[1] , Dr. G .Narsimha[2]

[1]Asst.Prof. Department of CSE, Methodist College of Engineering & Technology,Abids,Hyderabad,Telangana,500001,India.
rajshekardeva@gmail.com

[2]Vice-Principal & Professor, JNTUH, Sulthanpur, Sangareddy,
Telangana,India
narsimha06@gmail.com

ABSTRACT: The life risk factor for the cancer disease is becoming more and more in the present living life style. Many researchers focused on skin cancer using machine learning approaches, but when working with multi classification data, it is observed that ANN gives better results, if the input data is keeps on adding in the real time scenario. This proposed system focuses on the classification of the skin cancer types by designing the neural network based on the customized activation function. Many parameters are involved in designing a suitable classifier using NN like learning rate, optimizer, activator, and other normalization layers. This paper majorly focuses on the activation function and number of neurons associated with the layer because these two parameters play a vital role in the entire accuracy of the model. Among the existing activators, the combination of tanh and relu has given high value base on them; a new activation function is designed.

Keywords: Multi Classification, tanh, relu, estimators, optimizer, sparse categorical loss, cross entropy

INTRODUCTION:

The existing skin cancer detection system using neural network has used either same activation function for all the layers or different activation functions for different layers but the proposed system identifies the deviation of the input values at different points and designed a mathematical function that normalizes the inputs units[6]. The process of converting the dot product sum obtained from input into desired output form is known as “activation or transfer function”. The derivation computation of the activation function from the input values helps the back propagator to adjust the weights to minimize the error.

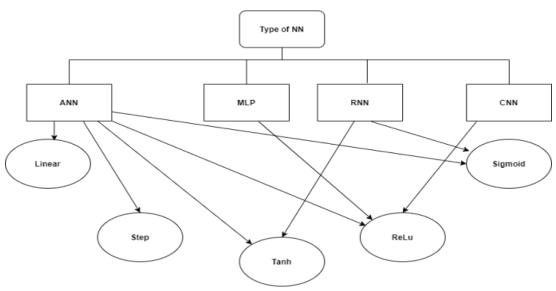

Figure 1: Common Activation Functions Implemented in Hidden Layers

In general, any neural network contains input layer, hidden layer, and output layer. Regarding output layer based on problem statement only 3 activation functions are available but when it comes to hidden layer, figure 1, represent the common activation functions utilized by different networks. The major goal of the tanh function is to adjust the values in between -1 to 1. Generally plotting these values on input scale of -10 to 10, the function produces an s-shaped graph[7]. The mathematical representation of this function is presented in equation (1)

$$\tanh(input) = \frac{(e^{input} - e^{-(input)})}{(e^{input} + e^{-(input)})} \text{ - (1)}$$

Tanh is famous for uniform weight initialization process but it suffers from “vanishing gradient problem”, where the activation function tries to converge the large input vector space into small range of values by using derivates, during this process, at some point saturation occurs and the value becomes zero. ReLu normalizes the values either to 0, if it is positive else gives the value as 1[8]. The function appears to be linear but it learns the complex function rapidly because of its linear units associated with the previous layers. The model suffers from “Dying ReLu problem” when most of the input values are normalized

JOURNAL OF ALGEBRAIC STATISTICS
Volume 13, No. 1, 2022, p. 511-516
https://publishoa.com
ISSN: 1309-3452

to 0. The mathematical notation for this is shown in equation (2)

$$ReLu(input) = \max(0, input) \text{ - } (2)$$

At this point, further applying derivation will not have any effect on model. The model addresses, this problem by applying the derivative based on the input value taken.

LITERATURE REVIEW:

In [1] Sourabh Shastri et al proposed Grading integrated with ensemble algorithm with both dynamic and static base filters applied to identify the important features. The model trains the system on different parameters and considers the one with majority voting. For each Meta classifier it generates rules using decision tree to identify the optimal features. The model divides the data into 8 batches and on each batch it applies a base classifier with cross validation mechanism. Out of these, 5 best classifiers are identified and passed to the next phase, which grades the algorithms as static or dynamic then using the ADABOOST; a tree is constructed from each batch and predictions of all these combinations are considered and the best combination is considered as "Best Set Builder". To identify the best combinations the model has computed Matthew correlation as similarity index.

In [2] Abdullah et al, designed reliefF algorithm to couple the features to identify the best features and uses conventional KNN to forecast the disease. The relief algorithm, computes rewards and penalties based on the prediction of classes. If the predicted classes belong to same class then it assigns rewards and proceeds further. If the predicted classes belong to other class then it assigns penalty and adjusts the weight that minimizes the distance between them. Each feature is assigned rank based on their weights. One with minimum weight is given top rank and with maximum is given least rank. KNN algorithm is implemented to classify the 74 samples available in the data set using the Euclidean distance.

In [3] Kun Xiangn et al implemented an elimination strategy on neurons based on their weights. The model constructs a DCNN to fine tune the features by attaching a threshold value to the last layer of the CNN; this reduces the number of connections in between the layers. The model retrains the system to reduce the learning rate so that the model preserves the significance of the features and this finally solves the problems associated with gradient descent. In this model, CNN tries to preserve all the edges by reducing the noise by finding the abnormal values and weights which are have very high positive values. Then the model uses percentage pruning mechanism to keep the restrictions and eliminates the values which are greater than 1-P, where p is positive rate. Finally, the model performs hierarchical pruning using standard deviation to fit all the edges and pixels to standard values.

In [4] Savy et al, designed a model by breaking the transfer learning process into two steps, model may improve the notion of SVM in machine learning as transfer constituent SVM (TrCSVM) and AdaBoost as transfer AdaBoost (TrAdaBost). In the first stage, TrCSVM obtains both homogeneous and heterogeneous domain features in the provided space. The TrAdaBoost was chosen mostly for its ability to dynamically raise or reduce weights based on the inter domain or intra domain it is learning in the current environment. This ensemble algorithm incorrectly classifies any image by chance, then automatically adjusts the weights and correctly classifies it in the next epoch. The entire uniqueness here is in the importance assignment, i.e., for accessing similar domain attributes, the algorithms assign lower weight values, and for accessing dissimilar domain attributes, the methods assign higher weight values.

In [5] Arvind Kumar et al, utilized rough set mechanism to identify the clinical attributes by encoding the patterns associated with the different attributes during the training process. The model discovers the hidden patterns by generating the decision rules associated with the information. The model applies lower and upper bound approximation. A rough set is a method for identifying traits and patterns in the face of uncertainty. The rough set is based on the idea that each phenomenon is linked to some data, and the data processing is based on that assumption. The model utilizes, Deci(Info), which is considered an inverse decision of reduced features.

Table 1: Limitations of the Existing Approaches

S.No	Author	Algorithm	Merits	Demerits
1	Shastri	Grading Ensemble Algorithm	Construction of Decision	With the increase of dataset size

JOURNAL OF ALGEBRAIC STATISTICS
Volume 13, No. 1, 2022, p. 511-516
https://publishoa.com
ISSN: 1309-3452

			tree helps the node to identify the pure features	increases and applications of all combination of algorithms will make the process complicated
2	Abdullah	reliefF+KNN	Optimal features are selected based on their weights because of which only minimum features gets selected	KNN is a simple distance based metric which cannot handle the data with more noise
3	Xiangn	Pruning nodes using DCNN	In general, when CNN extracts the features, there is no guarantee that all the important images are preserved but this DCNN preserves because it performs step by step pruning	When the system accepts high quality images then most of the unimportant features also may be preserved
4	Savy	Transfer learning	Non linear SVM works with complex relations very easily	-
5	Arvind	Rough Set	Decision tree information generalizes the model	When depth of the tree increases the complexity of the model increases

PROPOSED METHODOLOGY: The proposed system focuses on the selection of designing of the activation function required for designing the multi classification neural network of the skin cancer by analyzing the pit falls associated with the existing activation functions. The brief architecture of the proposed module is shown in figure 2.

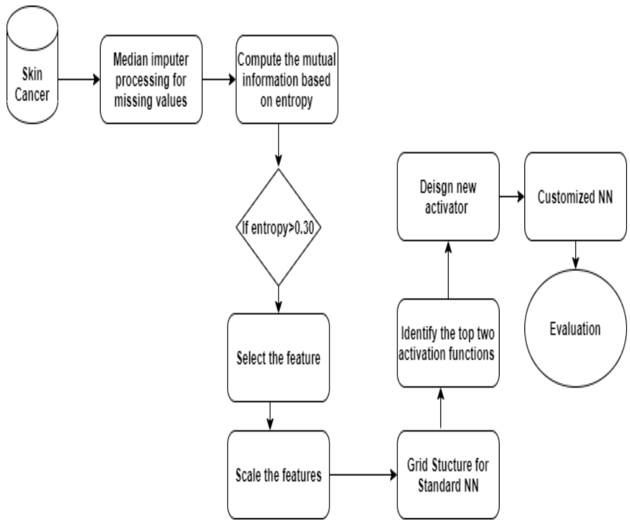

Figure 2: Customized Neural Network for Skin Cancer Classification

i. Pre-processing: The model analyzes the missing features in the dataset and found that a particular attribute values are skewed in nature, so it applies median as strategy in the imputer to clean the data.

ii. Feature Extraction: The dataset used in this model has 34 attributes, so to identify the important features it performs mutual information computation based on entropy characteristic. This metric considers two attributes and tries to reduce the uncertainty of one attribute based on relevant measure of other. The model tries to find the joint mutual information between attributes A1 & A2 as shown in equation (3)

$$Joint\_MI(A1, A2) = \iint P(A1, A2) * \log\left(\frac{P(A1,A2}{P(A1)*P(A2)}\right) d(A1)d(A2) \text{ - (3)}$$

The model has computed the joint mutual information between the attributes and class label. The model fixes the threshold value as 0.30 and considers the attributes whose Joint_MI value is greater than 0.30. Table 2 represents the values obtained by various attributes

Table 2: Joint Mutual Information Value of Attributes

Attribute Number	MI Value	Attribute Number	MI Value	Attribute Number	MI Value
4	0.31	14	0.38	24	0.50

	3		0		0
5	0.477	15	0.399	26	0.490
7	0.409	19	0.574	27	0.501
8	0.388	20	0.610	28	0.514
9	0.311	21	0.566	32	0.419
11	0.474	23	0.331		

These 17 features are considered as most import features and are passed as input to the ANN for performing multi classification.

iii. Multi Classification using ANN: The model designs a neural network as shown in figure(3) and the configuration of the neural network is illustrated in table 3.

Table 3: Configuration of Customized Neural Network

S.No	Estimator Name	Estimator Value	Description
1	Number of layers	1 input, 2 hidden, and 1 output	General strategy for ANN to start with 2 hidden layers
2	Number of neurons in the input and hidden layer	32	Geometric progression value of the half of the features
3	Number of neurons in the output layer	6	Number of class labels and number of neurons should be equal
4	Activation function for input and hidden layers	Customized	
5	Activation function for output layer	Softmax	Multi classification is able using softmax only
6	Optimizer	Adam	It has default learning rate with more productive tendency to train the features with less number of epochs
7	Loss Function	Sparse Categorical	Class label indices are treated as truth values

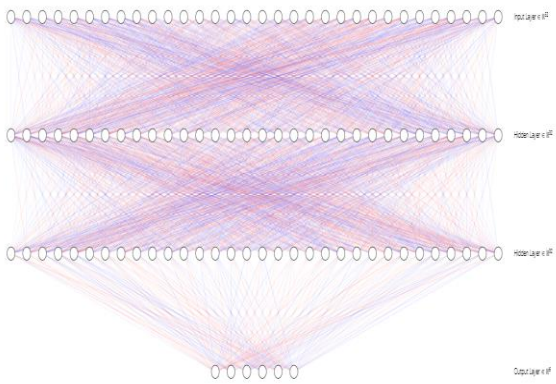

Figure 3: Represents Layered Architecture of Customized Neural Network

The Neural network contains 1 input layer, 2 hidden layers and one output layer. The neural network uses the dense layer architecture in which every node is connected to the every other node in the next layer. The hidden is designed by defining the activation function as shown in equation (4)

$$Skin\_Activator(input) = \frac{1}{1 + e^{input}}, \quad if\ input < 0$$

$$= 0, \quad \text{if input=0}$$

$$= -1+\frac{1}{input} \quad \text{if input > 0}$$

In between the layers, the model also uses dropout layer with a rate of 0.25 to normalize the values. In the neural network, the output layer is designed using "softmax" activation layer because the problem contains different class labels. The loss function is computed using "Sparse Categorical Cross Entropy",

JOURNAL OF ALGEBRAIC STATISTICS
Volume 13, No. 1, 2022, p. 511-516
https://publishoa.com
ISSN: 1309-3452

which is defined as shown in equation (5) and uses classes' indices as input.

$$Loss\_Sparse = -\,(\log(truth\_value)) \;-\; (5)$$

RESULTS & DISCUSSION:

The model compares the different accuracies of the neural network by changing the combination of activation functions for hidden layers; it maintains the first layer with tanh and last layer with softmax. The selection of the activation functions is done by considering the non linear units.

Table 4: Grid Results of Pre-defined Activation Functions

	Relu	sigmoid	tanh	Elu
Relu	93.46	88.5	95.3	90.77
Sigmoid	89.1	87.56	87	86.57
Tanh	87.43	86.80	87.25	87.97
Elu	88.3	88.11	88.5	88.98

Table 4 represents the accuracy of the model by applying combination of activation functions to the two layers of the neural network by iterating the model for 20 epoches and dividing the batches of size "32". Since, the combination of relu and tanh has obtained highest accuracy; these are utilized for further processing.

```
Epoch 91/100
10/10 [==============================] - 0s 2ms/step - loss: 0.1621 - accuracy: 0.9315
Epoch 92/100
10/10 [==============================] - 0s 3ms/step - loss: 0.1433 - accuracy: 0.9418
Epoch 93/100
10/10 [==============================] - 0s 3ms/step - loss: 0.1223 - accuracy: 0.9623
Epoch 94/100
10/10 [==============================] - 0s 2ms/step - loss: 0.1283 - accuracy: 0.9521
Epoch 95/100
10/10 [==============================] - 0s 2ms/step - loss: 0.1493 - accuracy: 0.9452
Epoch 96/100
10/10 [==============================] - 0s 2ms/step - loss: 0.1368 - accuracy: 0.9486
Epoch 97/100
10/10 [==============================] - 0s 3ms/step - loss: 0.1630 - accuracy: 0.9315
Epoch 98/100
10/10 [==============================] - 0s 2ms/step - loss: 0.1646 - accuracy: 0.9418
Epoch 99/100
10/10 [==============================] - 0s 2ms/step - loss: 0.1409 - accuracy: 0.9521
Epoch 100/100
10/10 [==============================] - 0s 2ms/step - loss: 0.1311 - accuracy: 0.9658
Average scores for all folds:
> Accuracy: 97.26776083310445 (+- 0.38639949735797574)
> Loss: 0.00712722862760226
------------------------------------------------------------------------
```

Figure 4: Sample of last 10 epochs along with Final Accuracy & Loss Results Presentation

Figure 4 represents the output for the last 10 epochs, the model has divided the data into 32 batch size and it is initialized to 100 epochs. The model has obtained an accuracy of "97.26%", with a minimum loss of 0.007%.

CONCLUSION:

Skin cancer is the most infectious disease but the death rate is less. Early identification of the cancerous cells helps the patients to treat with proper medication and prevents the further spreading of the cells. Traditional approaches utilized single activation functions for all the hidden layers but the proposed model has applied different combinations of the activation functions and identified the highest combination. The model designed a new activation by analyzing the pitfalls of ReLu and Tanh. The designed activation function is applied to both input and hidden layers and obtained an accuracy of 97.26%, which is greater than 95.3%. In the future work, the model tries to focus on the learning rate and optimizer customizations for the further improvement of the models.

REFERENCES:

[1] Shastri, S., Kour, P., Kumar, S. et al. GBoost: A novel Grading-AdaBoost ensemble approach for automatic identification of erythemato-squamous disease. Int. j. inf. tecnol. 13, 959–971 (2021). https://doi.org/10.1007/s41870-020-00589-4

[2] Alotaibi, A.S. Hybrid Model Based on ReliefF Algorithm and K-Nearest Neighbor for Erythemato-Squamous Diseases Forecasting. Arab J Sci Eng 47, 1299–1307 (2022). https://doi.org/10.1007/s13369-021-05921-z

[3] Xiang, K., Peng, L., Yang, H., Li, M., Cao, Z., Jiang, S., & Qu, G. (2021). A novel weight pruning strategy for light weight neural networks with application to the diagnosis of skin disease. Applied Soft Computing, 111, 107707. doi:10.1016/j.asoc.2021.107707

[4] Gulati, S., & Bhogal, R. K. (2019). Serving the Dermatologists: Skin Diseases Detection. Advances in Intelligent Systems and Computing, 799–822. doi:10.1007/978-981-13-7166-0_80

[5] Sinha, A. K., & Namdev, N. (2020). Feature selection and pattern recognition for different types of skin disease in human body using the rough set method. Network Modeling Analysis in Health Informatics and Bioinformatics, 9(1). doi:10.1007/s13721-020-00232-z

[6] Nauta M, Walsh R, Dubowski A, Seifert C. Uncovering and Correcting Shortcut Learning in

Machine Learning Models for Skin Cancer Diagnosis. *Diagnostics*. 2022; 12(1):40. https://doi.org/10.3390/diagnostics12010040

[7] David Wen, Saad M Khan, Antonio Ji Xu, Hussein Ibrahim, Luke Smith, Jose Caballero, Luis Zepeda, Carlos de Blas Perez, Alastair K Denniston, Xiaoxuan Liu, Rubeta N Matin, Characteristics of publicly available skin cancer image datasets: a systematic review, The Lancet Digital Health, Volume 4, Issue 1, 2022, Pages e64-e74, ISSN 2589-7500, https://doi.org/10.1016/S2589-7500(21)00252-1.

[8] A. Javaid, M. Sadiq and F. Akram, "Skin Cancer Classification Using Image Processing and Machine Learning," *2021 International Bhurban Conference on Applied Sciences and Technologies (IBCAST)*, 2021, pp. 439-444, doi: 10.1109/IBCAST51254.2021.9393198.

[9] Shoen, E. DermIA: Machine Learning to Improve Skin Cancer Screening. *J Digit Imaging* **34,** 1430–1434 (2021). https://doi.org/10.1007/s10278-020-00395-1