Predicting Breast Cancer using Convolutional Neural Network

Mr. Madhan S, Priyadharshuini P, Brindha C, Bairavi B
Department of Computer Science and Engineering
University College of Engineering Thirukkuvalai
Thirukkuvalai, Nagapattinam district,India

Abstract

Breast cancer is the second leading cause of death for women all over the world. But early detection and prevention can significantly reduce the chances of death. This paper deals with different statistical and analysis of breast cancer database for improving the accuracy in detection of breast cancer based on Convolutional Neural Network algorithm. The dataset is obtained from Wisconsin Hospital Madison. We analyzed the data and furthermore, these data can be validating , testing and trained. Finally the error histogram plotted form the dataset and we obtained the confusion matrix. So that the accuracy level can be predicted and get a higher accuracy which is up to 96% to 98%.

Keywords — *algorithms, disease diagnosis, Convolutional neural network.*

I. INTRODUCTION

A convolutional neural network (CNN or ConvNet) is one of the most popular algorithms for deep learning, a type of machine learning in which a model learns to perform classification tasks directly from images, video, text, or sound. CNNs are particularly useful for finding patterns in images to recognize objects, faces, and scenes. They learn directly from image data, using patterns to classify images and eliminating the need for manual feature extraction.

Applications that call for object recognition and computer vision such as self-driving vehicles and face-recognition applications rely heavily on CNNs. Depending on your application, you can build a CNN from scratch, or use a pretrained model with your dataset.

In deep learning, a convolutional neural network (CNN, or ConvNet) is a class of deep neural networks, most commonly applied to analyzing visual imagery.

Computer-aided diagnosis of breast cancer is potentially useful for reducing the numbers of grazes are missed by the radiologists at a reasonable cost. A convolution neural network (CNN) is used for classification of masses and normal tissue on mammograms.

Apart from other classifiers, Convolutional Neural Network and Multilayer Perceptron algorithms are also used for better prediction and accuracy. In our project, we have obtained 98.06% accuracy with 300 feature maps and 10 fold cross validation using Convolutional Neural Network (CNN). Our experiments not only have provided better results than the works mentioned above but also have indicated future scopes to do further research in the field of neural network based classification.

II. METHODOLOGY

A. *Dataset*

The dataset used in this project is obtained from the breast cancer database of the University of Wisconsin Hospitals Madison (Wolberg 1991). There are 11 attributes for each sample. Attributes 2 through 10 have been used to represent instances respectively. Number of instances is 699. But some of the instances are deleted due to missing attributes. There is a class attribute in addition to 9 other attributes. Each instance has one of 2 possibilities: Benign or malignant. One of the other numeric value columns is ID column of instances. Our dataset includes two classes as mentioned earlier. They are benign (B) and malignant (M). We further analyzed data and come up with total 30 attributes with 569 useful data.

The useful attributes are

1. Radius
2. Texture
3. Perimeter
4. Area
5. Smoothness
6. Compactness
7. Concavity
8. Concave points
9. Symmetry
10. Fractal dimension

B. *CNN-Based Feature Extraction*

ROI images were used as input to a pretrained CNN to extract CNN-based features. This CNN, i.e., the AlexNet, had been trained on the ImageNet dataset of 1.2 million high-resolution images and used to classify general objects into 1000 classes.The architecture of this pretrained CNN contained five convolutional layers, three pooling layers, and three fully connected layers. Given that the CNN was pretrained, our use of it was restricted to its original architecture and input image size of 227×227 pixels, and thus, 227×227 patches were extracted from the center of each 256×256 ROI. The output from the first fully connected layer, a vector of 4096 in length, served as the CNN-based features, which subsequently underwent dimension reduction by eliminating those features with zero-variance features across the datasets. The CNN-based features were then standardized with zero mean and unit variance prior to input to the classifier. The feature extractions were performed on a computer running openSUSE Linux operating system with 6-core/12-thread Intel Xeon CPU E5-2620 2.10 GHZ and 24GB memory.

III. PROCESS DESCRIPTION AND OBSERVATIONS

A. Pattern Recognition Network

Artificial neural networks are useful for pattern matching applications. Pattern matching consists of the ability to identify the class of input signals or patterns. *(fig .1)*

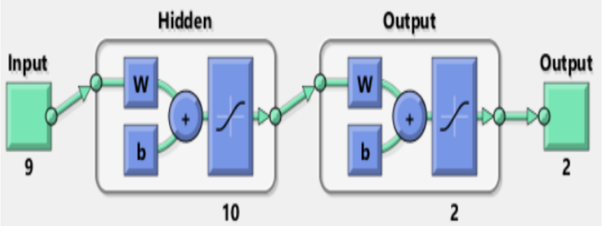


Fig 1. Pattern Recognition Network

B. Training, Validation, Testing

- The data used to build the final model usually comes from multiple datasets. In particular. Three data sets are commonly used in different stages of the creation of the model.
- The model is initially fit on a training dataset that is a set of examples used to fit the parameters of the model.
- In practice, the training dataset often consist of pairs of an input vector (or scalar) and the corresponding output vector (or scalar), which is commonly denoted as the target (or label). The current model is run with the training dataset and produces a result, which is then compared with the target, for each input vector in the training dataset. Based on the result of the comparison and the specific learning algorithm being used, the parameters of the model are adjusted. The model fitting can include both variable selection and parameter estimation.
- Successively, the fitted model is used to predict the responses for the observations in a second dataset called the validation dataset. The validation dataset provides an unbiased evaluation of a model fit on the training dataset while tuning the model's hyperparameters (e.g. the number of hidden units in a neural network). Validation datasets can be used for regularization by early stopping: stop training when the error on the validation dataset increases, as this is a sign of overfitting to the training dataset. This simple procedure is complicated in practice by the fact that the validation dataset's error may fluctuate during training, producing multiple local minima. This complication has led to the creation of many ad-hoc rules for deciding when overfitting has truly begun.
- Finally, the test dataset is a dataset used to provide an unbiased evaluation of a final model fit on the training dataset. When the data in the test dataset has never been used in training (for example in cross-validation), the test dataset is also called a holdout dataset.

C. Train the network

Plot trainstate (tr) plots the training state from a training record tr returned by train. *(fig .2)*

```
[net, tr] = train(net, inputs, targets)
```

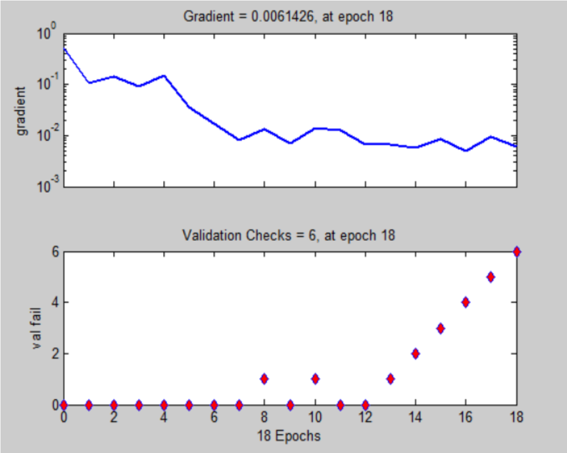


Fig 2. Training dataset

D. Test the network

1) Performance

Plot perform(TR) plots error vs. epoch for the training, validation, and test performances of the training record TR returned by the function train. *(fig . 3)*

```
outputs = net (inputs);
errors = gsubtract(target, outputs);
 performance = perform (net, targets, outputs);
```

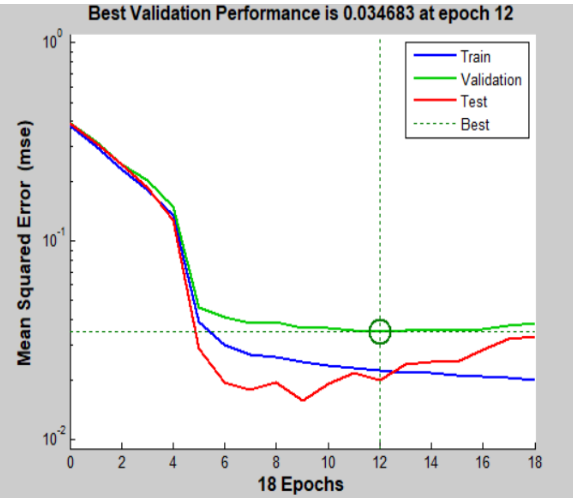


Fig 3. Performance

2) *Error Histogram*

- ploterrhist(e) plots a histogram of error values e.
- ploterrhist(e1,'name1',e2,'name2',...) takes any number of errors and names and plots each pair.
- ploterrhist(...,'bins',bins) takes an optional property name/value pair which defines the number of bins to use in the histogram plot. The default is 20. *(fig .4)*

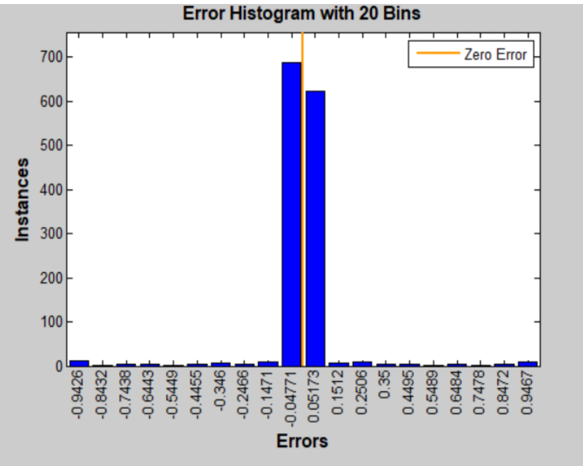
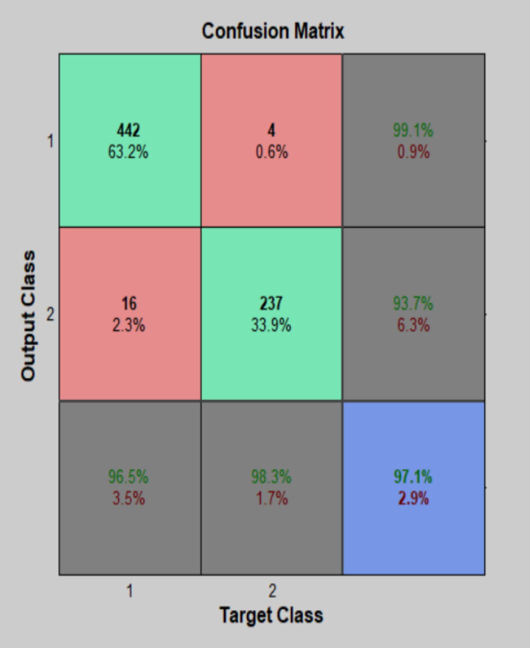


Fig 4. Error Histogram

3) *Confusion Matrix*

- Plotconfusion(targets,outputs) plots a confusion matrix for the true labels targets and predicted labels outputs. Specify the labels as categorical vectors, or in one-of-N (one-hot) form.
- On the confusion matrix plot, the rows correspond to the predicted class (Output Class) and the columns correspond to the true class (Target Class). The diagonal cells correspond to observations that are correctly classified. The off-diagonal cells correspond to incorrectly classified observations. Both the number of observations and the percentage of the total number of observations are shown in each cell.
- The column on the far right of the plot shows the percentages of all the examples predicted to belong to each class that are correctly and incorrectly classified. These metrics are often called the precision (or positive predictive value) and false discovery rate, respectively. The row at the bottom of the plot shows the percentages of all the examples belonging to each class that are correctly and incorrectly classified. These metrics are often called the recall (or true positive rate) and false negative rate, respectively. The cell in the bottom right of the plot shows the overall accuracy.
- plotconfusion(targets,outputs,name) plots a confusion matrix and adds name to the beginning of the plot title.
- plotconfusion(targets1,outputs1,name1,targets2,outputs2,name2,...,targetsn,outputsn,namen) plots multiple confusion matrices in one figure and adds the name arguments to the beginnings of the titles of the corresponding plots. *(fig . 5)*



	1	2	
1	442 63.2%	4 0.6%	99.1% 0.9%
2	16 2.3%	237 33.9%	93.7% 6.3%
	96.5% 3.5%	98.3% 1.7%	97.1% 2.9%

Fig 5. Confusion Matrix

IV. CONCLUSIONS

We have analyzed our data on the basis of Wisconsin Breast cancer database and we experimented with CNN classifiers and obtained highest accuracy. We did a deep investigation in the performance of different deep networks on this dataset. For deep networks, we have found that the convergence time significantly increases and it gets harder to optimize the network. In case of convolutional neural network. The same result might be obtained with different configuration of the network. Our results of CNN classifier (98.06% accuracy) show comparatively better performance in comparison the work of Karabatak and Cevdet-Ince (2009) [8] where the accuracy was 97.4% using Association Rules(AR) and Neural Network(NN). Such comparative analysis on breast cancer classification would provide further encouragement and insights on the efficient approaches for detection of cancer problems.

REFERENCES

[1] http://www.cancer.org/cancer/breastcancer/detaildguide/breast-cancer-key-statistics

[2] http://www.wcrf.org/int/cancer-facts-figures/data-specificcancers/breast-cancer-statistics

[3] https://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+(Original)

[4] Wolberg,William H., and Olvi L. Mangasarian, “Multisurface method of pattern separation for medical diagnosis applied to breast cytology.” Proceedings of the national academy of sciences 87.23(1990): 9193-9196.

[5] Zhang, Jianping, "Selecting typical instances in instancebased learning." Proceedings of the ninth international conference on machine learning. 1992.

[6] Angeline Christobel. Y, Dr. Sivaprakasam (2011), "An Empirical Comparison of Data Mining Classification Methods." International Journal of Computer Information Systems,Vol. 3, No. 2, 2011.

[7] Hong G, Nandi AK (2006), "Breast cancer diagnosis using genetic programming generated feature." Elsevier Pattern recognition 39: 980-987.

[8] Karabatak M, Ince MC (2009), "An expert system for detection of breast cancer based on association rules and neural network." Expert Systems with Applications 36: 3465-3469.