Machine Learning Engineer Nanodegree

Capstone Project

Akshit Gupta
April 25th, 2020

I. Definition

Project Overview

Breast cancer is the most common cancer in women worldwide, with nearly 1.7 million new cases diagnosed each year, representing about 25 percent of all cancers in women. It is the fifth most common cause of death from cancer in women, with an estimated 522,000 deaths (6.4 per cent of the total). Belgium had the highest rate of breast cancer in women, followed by Luxembourg. So it can help a lot of women to predict breast cancer based on certain features and attributes.

Problem Statement

The goal is quite straightforward, In this project, we will try to create a Breast cancer classifier; the tasks involved are as following:

- Collect and Process the Breast dataset.
- Develop a Classifier using Machine Learning Algorithms.
- Train the model to generate high accuracy.
- Test the classifier by feeding different types of tissues.

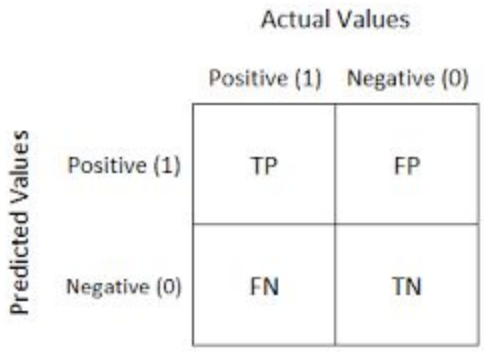
Metrics

1. Accuracy is a common metric for binary classifiers; it takes into account both true positives and true negatives with equal weight.
It is the ratio of number of correct predictions to the total number of input samples.

Link to dataset: https://www.kaggle.com/uciml/breast-cancer-wisconsin-data

$$\text{Accuracy} = \frac{\text{True Positives + True Negatives}}{\text{All Samples}}$$

2. Confusion Matrix is one of the most intuitive metrics used for finding the correctness and accuracy of the model. It is used for Classification problem where the output can be of two or more types of classes.



II. Analysis

Data Exploration

For this project we will use Breast Cancer Datset available on Kaggle by UCIML. Features are computed from a digitized image of a fine needle aspirate (FNA) of a breast mass. They describe the characteristics of the cell nuclei present in the image.

Attribute Information:

1) ID number

2) Diagnosis (M = malignant, B = benign)

3-32)

Ten real-valued features are computed for each cell nucleus:

a) radius (mean of distances from center to points on the perimeter)

b) texture (standard deviation of gray-scale values)

c) perimeter

d) area

Link to dataset: https://www.kaggle.com/uciml/breast-cancer-wisconsin-data

e) smoothness (local variation in radius lengths)

f) compactness (perimeter^2 / area - 1.0)

g) concavity (severity of concave portions of the contour)

h) concave points (number of concave portions of the contour)

i) symmetry

j) fractal dimension ("coastline approximation" - 1)

The mean, standard error and "worst" or largest (mean of the three largest values) of these features were computed for each image, resulting in 30 features. For instance, field 3 is Mean Radius, field 13 is Radius SE, field 23 is Worst Radius.

All feature values are recoded with four significant digits.

Missing attribute values: none

Class distribution: 357 benign, 212 malignant

	id	diagnosis	radius_mean	texture_mean	perimeter_mean	area_mean	smoothness_mean	compactness_mean	concavity_mean	concave points_mean	...
0	842302	M	17.99	10.38	122.80	1001.0	0.11840	0.27760	0.3001	0.14710	...
1	842517	M	20.57	17.77	132.90	1326.0	0.08474	0.07864	0.0869	0.07017	...
2	84300903	M	19.69	21.25	130.00	1203.0	0.10960	0.15990	0.1974	0.12790	...
3	84348301	M	11.42	20.38	77.58	386.1	0.14250	0.28390	0.2414	0.10520	...
4	84358402	M	20.29	14.34	135.10	1297.0	0.10030	0.13280	0.1980	0.10430	...

Fig 1: Data Snapshot

Exploratory Visualization

The plot below shows the count of Malignant and benign tissue in the database. The plot clearly shows that the benign tissue is in majority.

Link to dataset: https://www.kaggle.com/uciml/breast-cancer-wisconsin-data

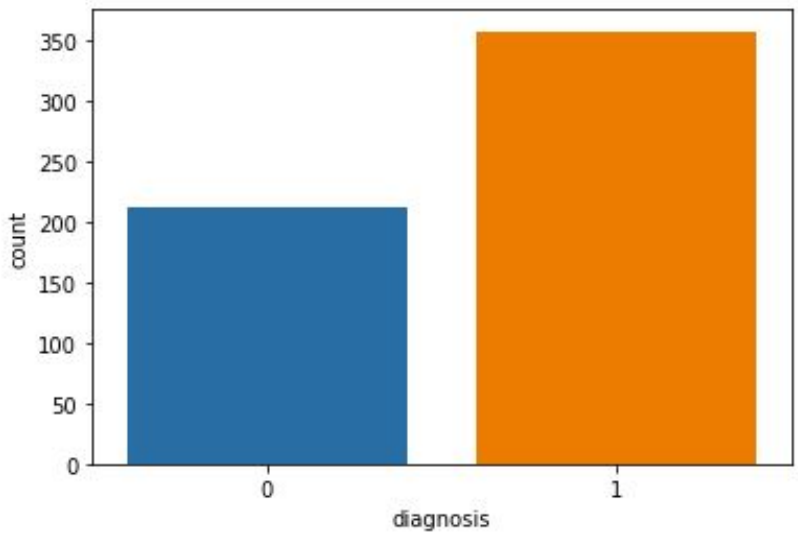


Fig 2: plot of malignant and benign tissue

Algorithms and Techniques

The project uses Keras technology to develop the classifier model.

Keras is a high-level neural networks API, capable of running on top of Tensorflow, Theano, and CNTK. It enables fast experimentation through a high level, user-friendly, modular and extensible API. Keras can also be run on both CPU and GPU.

The following parameters can be tuned to optimise the classifier:

❖ Training Parameter:

➢ Epochs

➢ Optimizer

➢ Loss Function

❖ Neural Network Architecture

➢ Number of Layers

➢ Layer Types

➢ Layer Parameters

Benchmark

To create a benchmark for my model, I trained the data on a Logistic Regression Classifier in scikit-learn library. This model generates an accuracy of 93.7%. The confusion matrix of the benchmark model is shown below:

Link to dataset: https://www.kaggle.com/uciml/breast-cancer-wisconsin-data

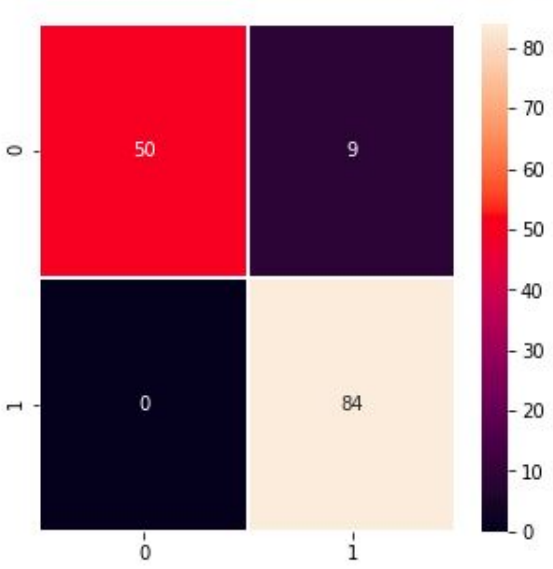


Fig 3 : Heatmap of confusion matrix of benchmark model

We can see the False positives and False Negatives contributes to the error of the model.
My goal is to improve the accuracy of the model from the given metrics.

III. Methodology

Data Preprocessing

Data is preprocessed in following steps:

● A target variable ‘category’ is introduced in the dataset with 0 as Malignant and 1 as Benign.

● The id column is removed as well the NaN values

● The data is normalized.

●The data is then split into testing and training sets.

Implementation

The model is implemented using Keras. A sequential model is developed with following layers and parameters:

1. Define the network architecture and training parameters.
2. Define the loss function, accuracy.
3. Define the input layer ,hidden layers and output layer
4. Train the model.
5. If the accuracy is not high enough, return to step 1.

Link to dataset: https://www.kaggle.com/uciml/breast-cancer-wisconsin-data

6. Save and freeze the trained network.

Summary of training of model can be seen below

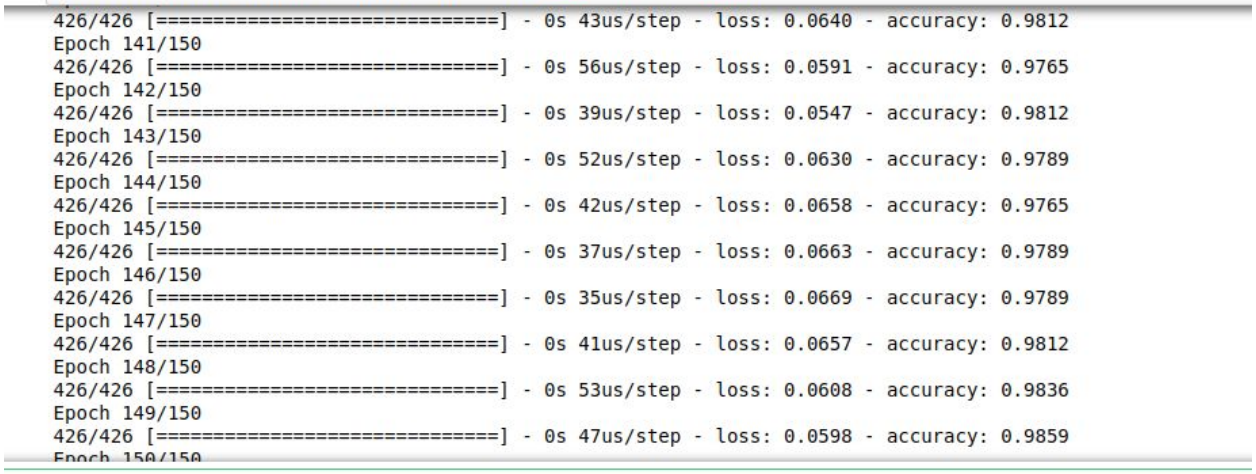


Fig 4 : Training Model

Refinement

As shown in the benchmark model, the accuracy of the model was calculated to be 93.7% which is quite good. The model created initially generated the accuracy of 90% which was further improved by introducing relu activation in hidden Layer and sigmoid activation in Output Layer. The layer parameters were also adjusted to generate the desired results.
The final Keras model was derived by training in an iterative fashion, adjusting the parameters. The final model has an accuracy of 99.2%.

IV. Results

Model Evaluation and Validation

The final architecture and hyperparameters were chosen because they performed the best among the tried combinations. As shown in fig 4, our model runs for 150 epochs.
The accuracy for our model came out to be 99.2%. The heatmap below shows 4 FP and 1 FN which shows our model has precision, recall and reliability.

Link to dataset: https://www.kaggle.com/uciml/breast-cancer-wisconsin-data

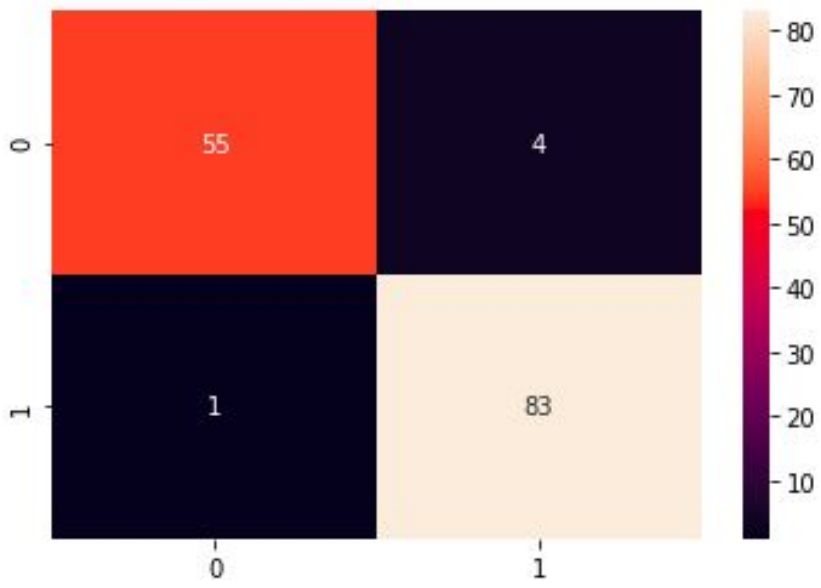


Fig 5: Heatmap of confusion matrix of proposed model

Justification

Our model evaluation metrics shows that the accuracy of the classifier was improved by approximately 5.5%. The evaluation model confusion matrix generated more FP and FN which led to lower precision and recall as compared to our classifier.

V. Conclusion

Reflection

The process used for this project can be summarised using the following steps:

1. Load the data
2. Clean the data and remove null values
3. Normalize the data
4. Divide the dataset in training and test
5. A benchmark was created for the classifier
6. Train the model and calculate the accuracy

I found step 6 most difficult because I had to familiarize myself with Keras and understand the documentation clearly I found using keras to train and artificial neural network quite interesting and would do more projects related to that.

Improvement

Since our model produced an accuracy of 99.2%, I think my model could be improved but it would not be an easy task as the accuracy is quite high.

Link to dataset: https://www.kaggle.com/uciml/breast-cancer-wisconsin-data