Predicting Cancer Malignancy with Limited Data

*Note: A Project Presented as part of the requirements for BIOF 509: Applied Machine Learning

Evaristus Mbanefo
Laboratory of Immunology
NEI, NIH
Bethesda, United States
evambanefo@yahoo.com

***Abstract*—Machine Learning has become very pivotal to Medicine. There are several tools that aid quick diagnosis and monitoring of treatment and prognosis. The entrance of machine learning has revolutionalized this approach. There is now a data driven approach that can predict presence or severity of disease based on data collected overtime from previous patients. This approach can even include data on drug treatment with good prognosis to direct best treatment. Using data collected from breast cancer patients, this tool is able to not only generate a model to predict breast cancer malignancy at over 0.95 accuracy. Above all, although this model used 31 variables, we show that the same result can be obtained at 0.92 accuracy using only 4 of the variables. This is important in situations where there is missing data or the use of equipment that do not collect all 31 variables for data collection.**

***Index Terms*—component, formatting, style, styling, insert**

I. Introduction

Machine Learning has come to stay and is becoming increasingly important in Medical practice. Machine learning tools take advantage of data to generate tools that aid quick diagnosis and monitoring of treatment and prognosis (Mostavi et al, 2015, Guo et al, 2015). In this era of personalized medicine (Haendel, 2018), this has become even more important. The entrance of machine learning has revolutionalized all aspects of medicine from social medicine up to neurosurgery, its application can be found in all aspects of health care.

There is increasingly data driven approaches that can be adopted (LeeCun et al, 2015) to take advantage of otherwise stored records to predict presence or severity of disease based on data collected overtime from previous patients. This approach can even include data on drug treatment with good prognosis to direct best treatment. Some studies have applied ML algorithms like convolutional neural networks for cancer prediction (Mostavi et al, 2015, De et al, 2013, Lyu et al, 2018).

Here, we use data collected from breast cancer patients, this tool is able to not only generate a model to predict breast cancer malignancy at over 0.95 accuracy. Above all, although this model used 31 variables, we show that the same result can be obtained at 0.92 accuracy using only 4 of the variables. This is important in situations where there is missing data or the use of equipment that do not collect all 31 variables for data collection.

II. Method

A. The data

Machine learning takes advantage of otherwise stagnant data or information in addition to continuous data collection. This results in enormous conservation of resources which can be focused on other aspect of the health care. The data utilized in this project is extracted from breast cancer images. Several features were extracted from the slides and converted into .csv file for easy processing. The data is available freely online at scikit-learn.

The variables are: 'mean radius', 'mean texture', 'mean perimeter', 'mean area', 'mean smoothness', 'mean compactness', 'mean concavity', 'mean concave points', 'mean symmetry', 'mean fractal dimension', 'radius error', 'texture error', 'perimeter error', 'area error', 'smoothness error', 'compactness error', 'concavity error', 'concave points error', 'symmetry error', 'fractal dimension error', 'worst radius', 'worst texture', 'worst perimeter', 'worst area', 'worst smoothness', 'worst compactness', 'worst concavity', 'worst concave points', 'worst symmetry', 'worst fractal dimension'. The classification were either Malignant or Benign.

III. Data Exploration

We first explored the data, converting into DtaFrame for easy preprocessing. For preprocessing, we used pandas, numpy, seaborn and matplotlib.pyplot libraries. After examining the data and exploring missing data, we confirmed the integrity of the data and visualised the head and tail. We plotted histogram of the two classes, in addition to the combined data. This enabled us to use the distribution of the data to decipher the variables that impacted the prediction more robustly.

The most significant parameters were mean radius, mean perimeter, mean area and mean concavity (Fig 1 - Fig 4) which were very efficient in distinguishing between the malignant and benign cases. We also show examples of variables that showed poor differential between the cancer severity categories, namely: mean fractal dimension, mean symmetry, mean smoothness and mean texture.

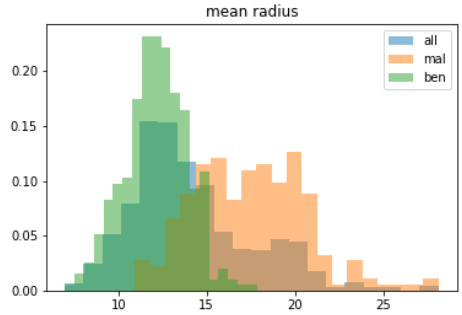

Fig. 1. Mean Radius (good correlation)

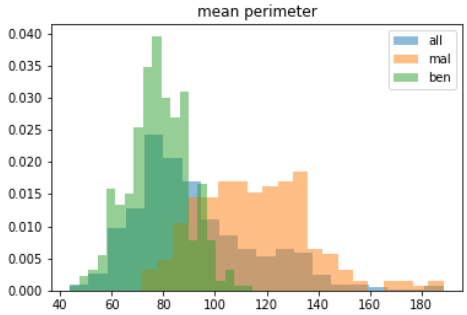

Fig. 2. Mean Perimeter (good correlation)

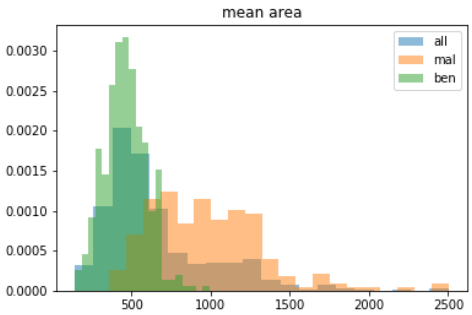

Fig. 3. Mean Area (good correlation)

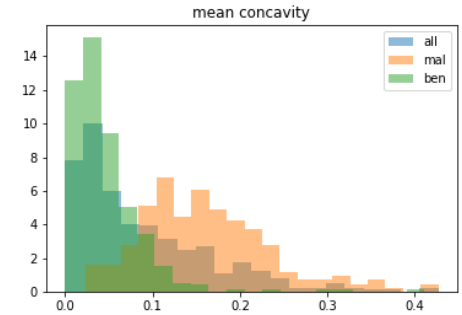

Fig. 4. Mean Concavity (good correlation)

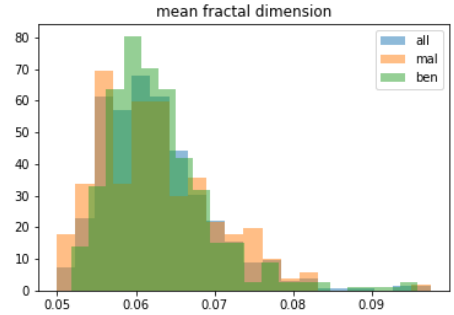

Fig. 5. Mean Fractal Dimension (poor correlation)

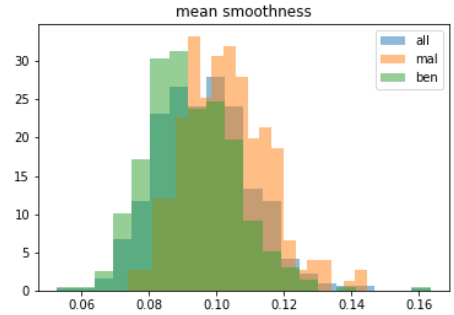

Fig. 6. Mean Smoothness (poor correlation)

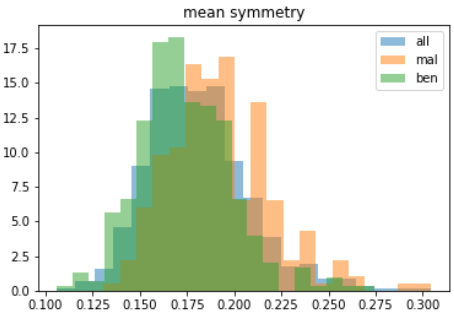

Fig. 7. Mean Symmetry (poor correlation)

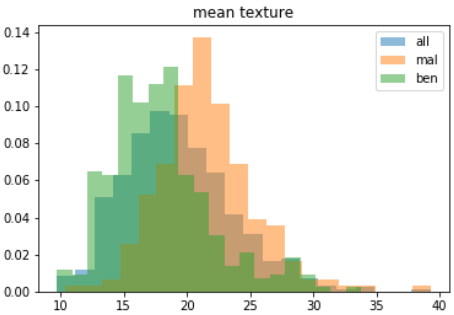

Fig. 8. Mean Texture (poor correlation)

IV. Modeling

A. Objective

Some of the variables available for a predictive model of cancer were of little value for the model. For instance, while radius, perimeter, area, concavity and concave point showed appreciable potential to differentiate breast cancer severity, the other variables were of less value in any predictive model from this dataset.

Moreso, the distribution of observations within the useful variables showed significant overlap between benign and malignant disease. We propose to generate an alternative predictive model using these variables with good correlations and compare the performance to model built using all the variables. We will then make a program that accepts few inputs from user to provide a predicition of cancer severity..

B. Label Encoding and Splitting

We used LabelEncoder and train test split functions in sklearn preprocessing and sklearn model selection, respectivly to randomly assign data to either training sets or testing set. The training set will be used for the training while the testing set will be used for validation.

C. Scaling and Standardization

We used the StandardScaler function also from sklearn package to scale both the x and y data in preparation for learning. The code blocks are in the Final project codes in github.

D. Model Selection

For the model selection, we imported both the Linear LogisticRegression, Support Vector Matrix and the k-means nearest neighbour models. The models were applied with optional parameters and thenn model fitting was performed, ready for prediction. An example of the code block can be seen below:

$$fromsklearn.neighborsimportKNeighborsClassifier \tag{1}$$

$$BC_classifier2 = KNeighborsClassifier(...) \tag{2}$$

$$BC_classifier2.fit(Variables_train, Diagnosis_train) \tag{3}$$

E. Prediction and Confusion Matrix

Next we used the test dataset to perform predictions, exported the result of the prediction to a DataFrame before calculating teh confudion matrix. Confusion matrix will use th enumber of correct and incorrect predictions to calculate the accuracy of the model. Based on our model and subsequent prediction, accuracy of over 0.958 our of 1 was observed. The k-means model returned o.951 accuracy.

Correctly predicted malignant: 50 Incorrectly predicted malignant: 3 Correctly predicted benign: 87 Incorrectly predicted benign: 3 Total of correct prediction: 137 Total of incorrect prediction: 6 Performance: 95.8041958041958

F. Model and Prediction based on fewer variables

We found earlier that a few variables impacted the model more than others. We attempted to repeat the prediction using only these four variables. Interestingly, the model still returned up to 0.9231 accuracy.

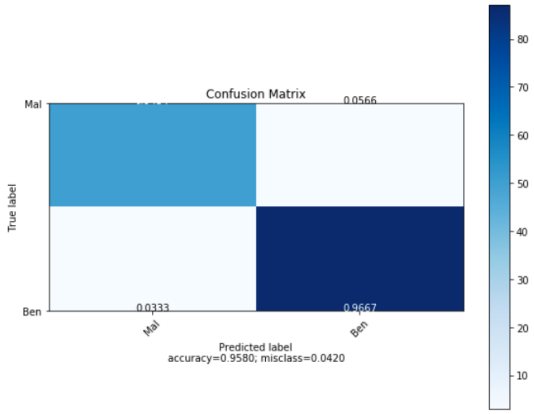

Fig. 9. Confusion Matrix for Model with 4 variables (0.958 accuracy)

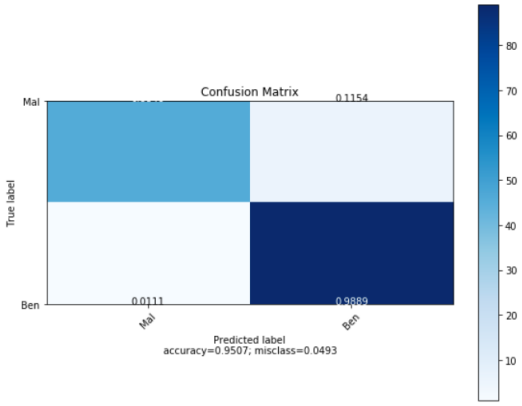

Fig. 10. Confusion Matrix for Model with K means (0.9507 accuracy)

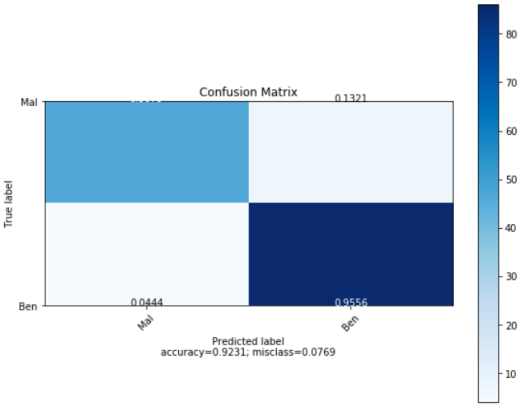

Fig. 11. Confusion Matrix for Model with 4 variables (0.9231 accuracy)

V. User Interface (Future prospect and Conclusiom)

We envisage future application of this model in the clinic, including in the case of incomplete data. Our tool will prompt user to enter values for variables starting with the key variables and will readily predict severity based on as few as the 4 major variables: mean radius, mean parameter, mean area, and mean concavity.

```
[321]: # Receive input
def BC_pred(ID:int):
    """
    This function collects patient data,
    transforms the data and predict the
    patient's cancer severity based on the
    pathology findings: radius, perimeter,
    area, concavity.

    input: patient ID
    Secondary input: variables as prompt inputs
    Output: Binary 0: malignant, 1: benign
    """

    #Collect patient data as prompt
    r=float(input("Enter radius (we expect 0-30): "))
    p=float(input("Enter perimeter (we expect 0-300): "))
    a=float(input("Enter area (we expect 0-3000): "))
    c=float(input("Enter concavity (we expect 0-0.5): "))
    new_Vars=[r,p,a,c] #make a list
    new_Vars=np.array(new_Vars) #convert to array

    # Get prediction for input, but first transform
    new_Vars = sc.transform(new_Vars.reshape(1, -1)) #transform
    new_Diag = BC_class4.predict(new_Vars) #predict severity

    #lets change teh binary to string for report purposes
    if new_Diag==0:
        new_Diag="malignant"
    else:
        new_Diag="benign"
    print("Predicted Cancer Severity for patient No:", ID, "is", new_Diag)

[*]: BC_pred(1)
Enter radius (we expect 0-30):
```

Fig. 12. User Interface

VI. Conclusiom)

Our work will contribute to quick diagnosis and classification of cancer. Such tool will still be useful, even in the situation of missing data since our algorithm did not need the 30 variables for accuracy. The user interface is a work in progress and can be further developed with other intuitive features to both make it amenable and accurate for clinical application.

Acknowledgment

Grateful to Dr James Anibal, Christina T and FAES.

References

[1] Milad Mostavi, Yu-Chiao Chiu, Yufei Huang and Yidong Chen, “Convolutional neural network models for cancer type prediction based on gene expression,” (2015). BMC Medical Genomics, 2020; 13 (Suppl 5): 44 DOI: 10.1186/s12920-020-0677-2

[2] Peng Guo; Koyel Banerjee; R. Joe Stanley; Rodney Long; Sameer Antani; George Thoma; Rosemary Zuna; Shelliane R. Frazier; Randy H. Moss; William V. Stoecker, Nuclei-Based Features for Uterine Cervical Cancer Histology Image Analysis With Fusion-Based Classification. IEEE Journal of Biomedical and Health Informatics (Vol 20, issue 6 pp. 1595-1607) DOI: 10.1109/JBHI.2015.2483318

[3] Haendel MA, Chute CG, Robinson PN. Classification, ontology, and precision medicine. N Engl J Med. 2018;379(15):1452?62.

[4] S. De, R. J. Stanley, C. Lu, R. Long, S. Antani, G. Thoma, et al., ”A fusion-based approach for uterine cervical cancer histology image classification”, Comput. Med. Imag. Graph., vol. 37, pp. 475-487, 2013.

[5] LeCun Y, Bengio Y, Hinton G. Deep learning. Nature. 2015;521(7553):436?44.

[6] Lyu B, Haque A. Deep learning based tumor type classification using gene expression data. In: Proceedings of the 2018 ACM International Conference on Bioinformatics, Computational Biology, and Health Informatics: ACM; 2018. p. 89?96.

00