BREAST CANCER DETECTION:

A MACHINE LEARNING ALGORITHM

Table of Contents

Introduction

- OBJECTIVE:
  - To build a Breast Cancer Classification system using Machine Learning with Python.

- Target:
  - The intended audience for this project are those who use Fine Needle Aspiration to detect whether an area of abnormal looking tissue or body fluid is a malignant or benign cancer.

Dataset

- The dataset has 596 entries and 31 columns
- Dataset was taken from Kaggle: https://www.kaggle.com/uciml/breast-cancer-wisconsin-data

- Columns:
  - Diagnosis (M = malignant, B = benign)
  - Radius mean = mean of distances from center to points on the perimeter
  - Texture mean = sd of gray-scale values
  - Perimeter mean = mean size of the core of tumor
  - Area mean
  - Smoothness mean = mean of local variation in radius lengths

- Compactness mean = mean of parameter^2 / area – 1.0
- Concavity mean = mean of severity of concave portions of the contour
- Concave points = mean for # of concave portions of the contour
- Symmetry mean
- Fractal dimensions = mean for “coastline approximation” – 1.0
- Radius se = standard error for the mean of distances from center to points on the parameter
- Texture se = standard error for standard deviation of gray-scale values
- Perimeter se
- Area se
- Smoothness se = standard error for local variation in radius lengths
- Compactness se = standard error for perimeter^2 / area – 1.0
- Concavity se = standard error for severity of concave portions of the contour
- Concave points se = standard error for # of concave portions of the contour
- Symmetry se
- Fractal dimension se = standard error for “coastline approximation” – 1.0
- Radius worst = largest mean value for mean of distances from center to points on the parameter
- Texture worst = largest mean value for standard deviation of gray-scale values
- Perimeter worst
- Area worst
- Smoothness worst = largest mean value for local variation in radius lengths
- Compactness worst = largest mean value for perimeter^2 / area -1.0
- Concavity worst = largest mean value for severity of concave portion of contour
- Concave points worst = largest mean value for # of concave portions of the contour
- Symmetry worst
- Fractal dimension worst = largest mean value for “coastline approximation” – 1.0

Libraries

- numpy
- Pandas = to create dataframes
- Sklearn.model_selection = to split dataset to training and test data
- Sklearn.linear_model = to initiate Logistic Regression model
- Sklearn.metrics = to calculate accuracy score

ANALYSIS

- Pre-processing:
  - Check to look for missing values in any of the 32 columns
  - Check for the distribution of the Target variable (M or B) called “label”:
    - There are 357 entries that were considered Benign and 212 that were Malignant.

- Splitting the Features and Target columns:
  - For the features, all but the “label” columns were included.
  - The Target data only included the column “label”

- Splitting the features and Target to training and test datasets.
  - The test dataset comprised of 20% of the features and target data
  - The rest went to the training data

- MODEL TRAINING
  - To train the Logistic Regression to be able to accurately predict the classification based on the feature columns, the model was trained on the training dataset.
- MODEL EVALUATION
  - To evaluate how well the model accurately predicted the classification on the training model, the accuracy score was calculated. The accuracy score was calculated based on how many classification the model got right and then compared it to the actual classification on the training target dataset.
  - The accuracy of the training data was 0.9516 or around 95%.
  - To ensure that the model will not heavily rely on the training data to based its classification on, the accuracy score of the testing data was also calculated. The score was 0.9298 or around 93%.
  - Since the accuracy score of the training and test dataset are pretty close, there is no case of overfitting in this model.

WHY Logistic Regression?

- LOGISTIC REGRESSION:
  - models the probabilities for classification problems with two possible outcomes.
  - Is used when the dependent variable is categorical.
  - Since a model that classifies the data points into two possible outcomes: M (malignant) or B (benign) is necessary, Logistic Regression is just one way to predict classification for the problem.

BUILDING A PREDICTIVE SYSTEM

- After training the model to accurately predict the classification of the training dataset and calculate the accuracy score, a predictive system can be constructed to be able to use the model.
- To build a predictive system that could be of use, it must be able to take input that consists of values pertaining to the dataset columns.
- But in order for the Logistic Regression model to be able to use the input data, it must first be reshaped as the model was trained to read multiple data points.
- The input data contains only a single data point and in order to be of use, it has to be changed to a numpy array. Then reshape the numpy array as there is only one data point to predict a classification for.
- After reshaping, the model can be used to classify whether the Breast cancer is malignant or benign.

CONCLUSIONS

- Detecting whether a body of tissue is malignant or benign using Fine Needle Aspiration has been made easier with Logistic Regression. With an accuracy score of 95% for the training dataset and 93% for the testing dataset, there is no issue of overfitting. The Logistic Regression model is good at predicting the classification. From this model, a predictive system can be built that can be used by institutions to diagnose a patient.