Aum Amriteswaryai Namaha



**Minor Project 21CSA697A**

**Final Report**

# Title: Minor project demo- Breast-Cancer-Classification-Using-Mammography-Masses

**Student Name:**  **Amrita Chaudhuri**

# Roll No.: AA.SC.P2MCA2107449

1. **Course Overview**

*(List the enrolled courses, and indicate whether completed or not)*

|  |  |  |
| --- | --- | --- |
| **Sl. No.** | **Courses** | **Outcomes**  *(Describe the learning outcomes achieved from each course)* |
| **1.** | **Structuring Machine Learning Projects by DeepLearning.AI Andrew NG** | * Explain why Machine Learning strategy is important * Apply satisficing and optimizing metrics to set up your goal for ML projects * Choose a correct train/dev/test split of your dataset * Define human-level performance * Use human-level performance to define key priorities in ML projects * Take the correct ML Strategic decision based on observations of performances and dataset |
| **2.** | **2. Advanced DataScience with IBM**   * **b. Advanced Machine Learning and Signal Processing IBM** * **c.** **Applied AI with DeepLearning** * **Fundamentals of scalable data science** | * Describe the challenges of data analytics * Describe different methods used for IoT data analysis * Create and deploy a test data generator capable of simulating IoT sensor data coming from a hypothetical washing machine * Apply a solution that captures and stores IoT data from connected devices with Node-Red and Apache ChouchDB NoSQL * Show how to process large data with ApacheSpark * Describe different statistical measures (moments) used in summarizing data * Describe how to process large amount of data arriving in high velocity by using ApacheSpark and SQL * Explain the concept of multi-dimensional vector spaces and how any type of data corpus can be understood as points in that space * Illustrate transformation and basic visualization of data * Linear algebra , ML pipeline , statistics , Regression , clustering, SVM, K means |

1. **Assessments Completed**

*(Any two in detail)*

|  |  |  |
| --- | --- | --- |
| **Sl. No.** | **Courses** | **Assessments- There were many assessments including both quiz and hands-on lab in IBM Watson studio. Here mentioning 2 assignments.**  **1**. generate sensor data-->https://github.com/IBM/coursera/raw/master/coursera\_ml/a2.parquet---> analysing the data using spark-->creating a VectorAssembler which consumes columns X, Y and Z and produces a column “features”\n"--> instantiate a classifier from the SparkML package and assign it to the classifier variable-->Rename the “CLASS” column to “label”-->Specify the label-column correctly to be “CLASS”-->train and evaluate-->submit  2. again load parquet file-->check if we have balanced classes – this means that we have roughly the same number of examples for each class we want to predict which is also important for classification and clustering -->create a VectorAssembler which consumes columns X, Y and Z and produces a column “features”--> insatiate a clustering algorithm from the SparkML package and assign it to the clust variable-->from clustering import Kmeans-->from pyspark import pipeline--> fit the model and predict  3.generate data with IoT data storage calculator for IBM cloud-->Create a test data generator using Node-RED  -->Publishing data to the Watson IoT platform-->Implement a flow to subscribe to this data and store it in Nosql DB[Apache CouchDB(Cloudant)] --> creating a function which is used to create a data frame from a cloudant data frame using the "DataSource" which is some sort of a plugin which allows ApacheSpark to use different data sources.-->then passed the dataframe object. also registered the dataframe in the ApacheSparkSQL catalog --> issue queries against the "washing" table using "spark.sql()-->implement a function which returns a (python) list of string values of the field names in this data frame--> connect to the cloudant database --> test the created functions |
| **1.** | **Course completion status snapshot** |  |
|  |  |  |

[*https://coursera.org/share/4fa8904863b27761093bbbcaec5ffde0*](https://coursera.org/share/4fa8904863b27761093bbbcaec5ffde0)

**

*(Details about the hands-on implementation)*

1. **Abstract of the project**

*(Write about the hands-on implementation in not more than 100 words)*

This project is about helping detecting the severity of breast cancer based on the state of their mammographic masses results (Benign or Malignant), using different machine learning algorithm which I learnt from Coursera . the algorithms used are SVM, Logistic Regression, Random Forest, Naive Bayes, Artificial Neural Network.

1. **Problem Statement** Mammography is the most effective method for breast cancer screening available today. However, the low positive predictive value of breast biopsy resulting from mammogram interpretation leads to approximately 70% unnecessary biopsies with benign outcomes. To reduce the high number of unnecessary breast biopsies, several computer-aided diagnosis (CAD) systems have been proposed in the last years. These systems help physicians in their decision to perform a breast biopsy on a suspicious lesion seen in a mammogram or to perform a short term follow-up examination instead.
2. **Methods/ Algorithms 🡪** Logistic Regression, Decision tree, SVM, neural network , random forest, KNN
3. **System Design**
4. **Software Requirements**
5. Python
6. Pandas
7. matplotlib
8. numpy
9. scikit-learn
10. Tensorflow
11. Keras
    1. **Hardware Requirements**
    2. **DATASET :** <https://archive.ics.uci.edu/ml/datasets/Mammographic+Mass>

**DESCRIPTION:** This data contains 961 instances of masses detected in mammograms, and contains the following attributes:

1. BI-RADS assessment: 1 to 5 (ordinal)
2. Age: patient's age in years (integer)
3. Shape: mass shape: round=1 oval=2 lobular=3 irregular=4 (nominal)
4. Margin: mass margin: circumscribed=1 microlobulated=2 obscured=3 ill-defined=4 spiculated=5 (nominal)
5. Density: mass density high=1 iso=2 low=3 fat-containing=4 (ordinal)
6. Severity: benign=0 or malignant=1 (binominal)

The file mammographic\_masses.data.txt contains the attributes data. The file mammographic\_masses.names.txt gives the description of the attributes.

BI-RADS is an assesment of how confident the severity classification is; it is not a "predictive" attribute and so we will discard it. The age, shape, margin, and density attributes are the features that we will build our model with, and "severity" is the classification we will attempt to predict based on those attributes.

Although "shape" and "margin" are nominal data types, which sklearn typically doesn't deal with well, they are close enough to ordinal that we shouldn't just discard them. The "shape" for example is ordered increasingly from round to irregular.

* 1. **Design of the System**
  2. **Module Description**

**Tensorflow- 2.11**

**Pandas 1.5.3**

**Numpy 1.24.2**

**GITHUB LINK for Jupyter notebook**

[**https://github.com/Amritaac/Breast-cancer-miniProject/blob/main/MCA\_AI-Sem3-Mammography-classification-AmritaChaudhuri%20(2).ipynb**](https://github.com/Amritaac/Breast-cancer-miniProject/blob/main/MCA_AI-Sem3-Mammography-classification-AmritaChaudhuri%20(2).ipynb)

1. **System Implementation**

The top 5 rows of our data show that we need to add column names as well as to deal with "?"

if there are null values and drop them

Checking how each of the features is distributed

As there are different scales for the features and some of the algorithms require a prior normalization of input data, need to normalize using StandardScaler

In order to train and test our data, let's split it into two parts: 75%-25%. Then we'll apply the above mentioned models to see which of them performs better with selected parameters.

using DecisionTreeClassifier to fit training data

Print corresponding score and confusion matrix.

Using GraphViz plot Decision trees

Plotting ROC curve

Implementing K-fold cross validation upto k=10

Applied RandomForestClassifier and check accuracy score

Then KNN. KNN-using neighbors.KNeighborsClassifier - Starting with a K of 10. K is an example of a hyperparameter - a parameter on the model itself which may need to be tuned for best results on a particular data set.

As choosing K is a bit tricky, so trying different values of K. Writing a for loop to run KNN with K values ranging from 1 to 50 and see if K makes a substantial difference.

Then applying naive\_bayes.MultinomialNB

Next applying SVM with different kernels and check ROC.

After that LogisticRegression

Last applied ANN with KERAS

1. **Results and Discussions**

By comparing all classifiers algorithms, It has been observed that top accuracy gained with deep learning algorithm with AUC of 0.87 +/- 0.01. Next is Logistic Regression with AUC of 0.847 +/- 0.01 and SVM (Linear Kernel) with AUC of 0.842 +/- 0.01

**Conclusion**

We may try to use grid search to find out a combination of hyperparameters to help improve the accuracy further for this particular project.

|  |
| --- |
| **Date 8th June 2023** |
| **Student Name and Signature** |