**LUNG CANCER DETECTION**

CS 6350 – Final Semester Project

Big Data | 1st December, 2017

**Team members:**

|  |  |
| --- | --- |
| **Sangeeta Kadambala** | **sxk160731** |
| **Gautam Gunda** | **gxg161830** |
| **Lavanya Swetha Gudimetla** | **lxg161730** |
| **Mohanapriya Narapareddy** | **mxn160330** |

**Table of Contents**

[**INTRODUCTION AND PROBLEM DESCRIPTION** 3](#_Toc499937493)

[**RELATED WORK** 3](#_Toc499937494)

[**DATA DESCRIPTION** 3](#_Toc499937495)

[**DATA DISTRIBUTION** 5](#_Toc499937496)

[**PRE PROCESSING TECHNIQUES** 6](#_Toc499937497)

[**PROPOSED SOLUTION AND METHODS** 9](#_Toc499937498)

[**EXPERIMENTAL RESULTS AND ANALYSIS** 10](#_Toc499937499)

[**CONCLUSION** 13](#_Toc499937500)

[**CONTRIBUTIONS** 13](#_Toc499937501)

[**REFERENCES** 14](#_Toc499937502)

# **INTRODUCTION AND PROBLEM DESCRIPTION**

In the recent years, lung cancer has become the most severe and has been reported to have more deaths compared to other cancer types. According to SEER, in US, someone dies every 3.3 minutes due to lung cancer. It kills more people compared to breast, colon, and prostate cancer combined.

Presently, just 16% of lung cancer cases are diagnosed at an early stage. Clinical researchers are working for better techniques to screen high-risk groups earlier. A 3-D scan of the lungs, called a CT scan, detects the presence of cancer more effectively than the traditional x-rays. Through these scans, trained radiologists can detect the signs of lung cancer early and save lives. However, it has high false positive rate. i.e the test reports cancer when it’s not. The false positive result in unnecessary invasive follow-up procedures, financial burden, and worry for patients.

Our target is to accurately detect early stages of cancer through data entered by radiologists by seeing CT scans.

# **RELATED WORK**

We analyzed the existing research literature for work on both cancer prediction and nodule detection/classification over CT scans and other forms of medical imagery.

Our training is based on the coordinates of cancerous cells provided by the radiologists. The paper on “Lung Cancer Detection on CT Images by Using Image Processing” [1] says that more round and opaque the mass is, higher is the chance of it being a nodule (cancerous mass). We based our feature extraction on this. Thus, if the eccentricity of the mass <1, it will have lower chance of being cancerous.

As per the document" Predicting Lung Cancer Incidence from CT Imagery "[2] use of CNN in cad processing helps in detection of large masses. These techniques give a good accuracy on large masses. We will be using a similar model but vary the filters and the activation functions to generate a signal which will be sensitive to smaller masses (nodules<3mm) and thus aid in early detection of cancer.

# **DATA DESCRIPTION**

**Data Collection:**

The Lung Image Database Consortium image collection (LIDC-IDRI) contains computed tomography (CT) scans with marked-up annotated lesions of lungs. This data set contains information about 1018 cases. The associated xml file represents the results of two-phase image annotation process, which is performed by four experienced thoracic radiologists. Image annotation process is done in two phases as follows:

1. **Initial blinded-read phase:**

In this phase each radiologist independently reviewed each CT scan and marked lesions belonging to one of following three categories based on nodule size.

1. **Unblinded-read phase:**

In this phase each radiologist independently reviewed their own marks along with the anonymized marks of the three other radiologists and conclude with final results.

For a specific CT series, the results from the unblinded reading session from the four independent radiologists are presented in a single xml file.

Data Description:

Nodule represents a spectrum of abnormalities. A lesion will be considered a “nodule” if it satisfies the definition of nodule i.e. should conform to nodular morphology and visual nodule library. The three nodule categories are as follows:

1. **Nodules >= 3mm diameter (but < 30mm diameter):**

The reader marks the outline of each nodule and marks the regions of exclusion. He considers several characteristics based on their appearance and internal features. For each nodule, he reports the characteristics on the scale of 1-5. The characteristics are as follows: Subtlety, Internal structure, calcification, sphericity, margin, speculation, texture and malignancy.

1. **Nodules < 3mm diameter:**

Each nodule is represented by an approximated three-dimensional center-of-mass. Assessment is not performed for these nodules characteristics.

1. **Non-Nodules >= 3mm diameter:**

Each nodule is represented by an approximated three-dimensional center-of-mass and also no further assessment of its characteristics.

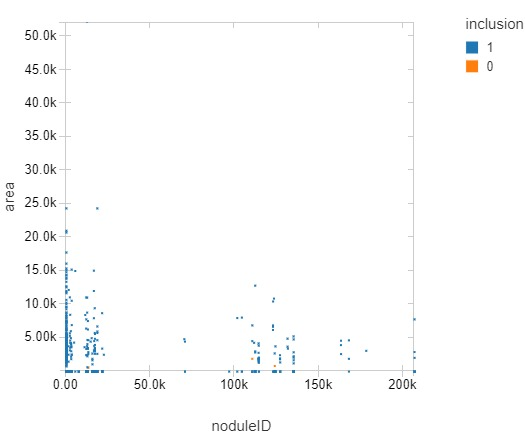
Below is the structure of the XML files that’s provided:

|  |
| --- |
| <LidcReadMessage>  <ResponseHeader>  <SeriesInstanceUid></SeriesInstanceUid>  </ResponseHeader>  <readingSession>  <unblindedReadNodule>  <noduleID></noduleID>  <characteristics>  <subtlety></subtlety>  <internalStructure></internalStructure>  <calcification></calcification>  <sphericity></sphericity>  <margin></margin>  <lobulation></lobulation>  <spiculation></spiculation>  <texture></texture>  <malignancy></malignancy>  </characteristics>  <roi>  <inclusion></inclusion>  <edgeMap><xCoord></xCoord><yCoord></yCoord></edgeMap>  </roi>  </unblindedReadNodule>  </readingSession>  </LidcReadMessage> |

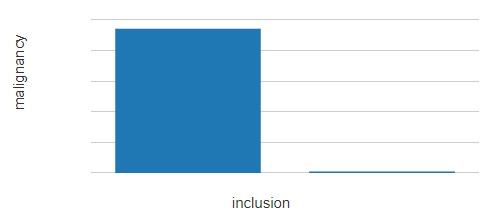
# **DATA DISTRIBUTION**

Our dataset consists of 124 GB of CT scan images. We used the corresponding xml files for those images which consist data regarding nodules. The xml files consist of the structural characteristics and information of the nodules by considering various factors. They contain the boundary coordinates of all the nodules greater than 3mm, centroid coordinates of nodule less than 3mm and centroid coordinates of non-nodule.

The below graph represents the calculated area distribution of the various nodules. The blue dotted ones represent that the inclusion factor is true.



If we consider the malignancy factor and inclusion factor, and if the inclusion is true, malignancy is high. If the inclusion is false, malignancy is very low on the scale of 0 to 6. So, from here we can observe the importance of this feature for cancer prediction.



# **PRE PROCESSING TECHNIQUES**

**Text Data Preprocessing:**

The XML data is converted in to the proper data frame.

1. At first we consider ***<LidcReadMessage>*** as the main row tag which is the unique XML ID number to represent each patient.
2. And then we define two struct fields for ***ResponseHeader*** and ***readingSession*** under the main row tag.
3. In ***ResponseHeader*** we form one more struct field named ***SeriesInstanceUID***, which is the unique ID of the particular CT series linked with the XML file.
4. In ***readingSession*** we form a loop and define a struct field named ***unblindedReadNodule***, which is the section that contains all the information about each Nodule.

In this struct field we form a loop again and define three struct fields, ***noduleID***, ***characteristics*** and ***roi***.

We obtain the following characteristics:

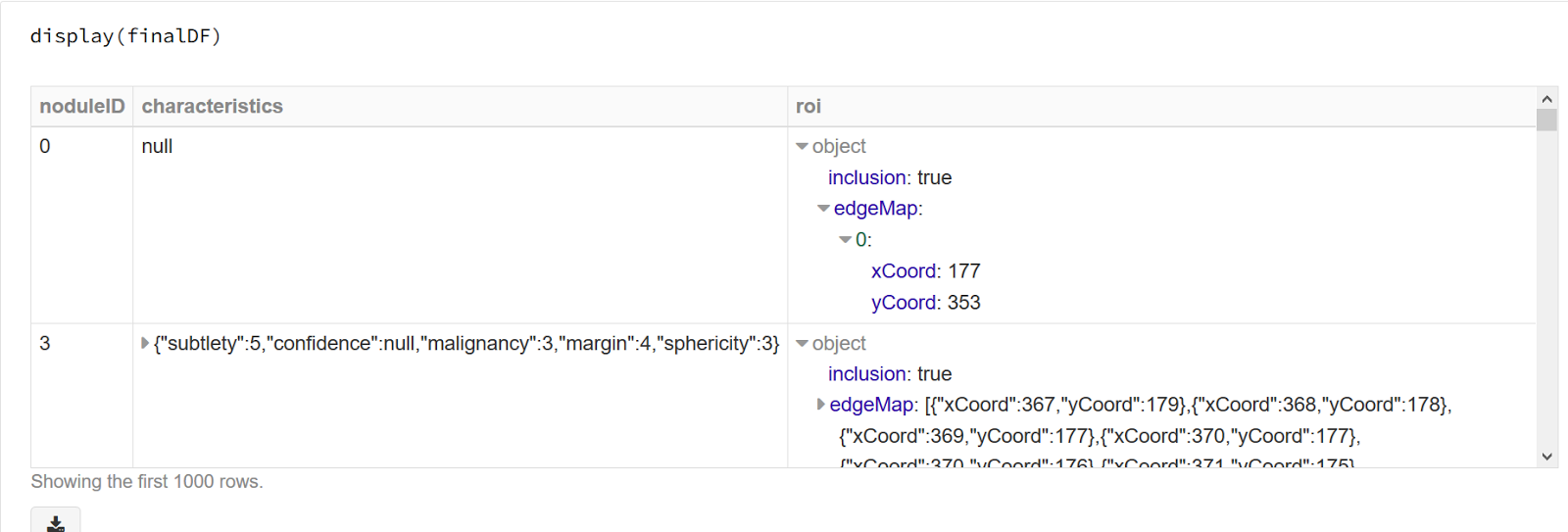
1. Subtlety - It is the measure of the level of difficulty in detection.
2. Internal structure - The internal structure represents the internal composition of the node which includes various elements like soft tissue, fluid, and air.
3. Calcification- It represents the amount of calcium deposition in the node.
4. Sphericity- It is the outer 3-D shape of the nodule.
5. Margin- It gives the description about the margins of the nodule.
6. Lobulation- This is the factor associated with the malignancy and measures the extent of infected nodules.
7. Spiculation- It measures the amount of spiculation the specific node contains.
8. Texture - It represents the internal composition of the nodule including various components in the nodule.
9. Malignancy - It measures the presence of the malignant tumour.

We consider all the characteristics to identify the presence of cancer and define nine struct fields which define the various characteristics of the nodule.

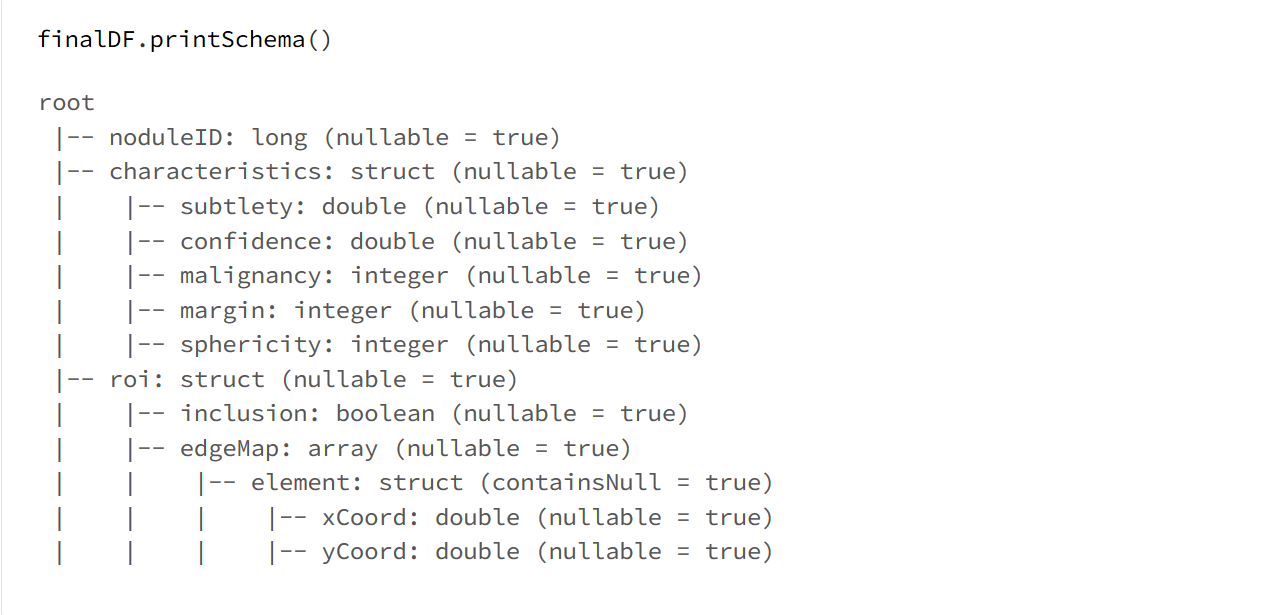
We also form the loop over the roi and define various struct fields like the

1. Edgemap -This represents the x, y coordinates determined by the given Z position. We can locate the outline of nodule with this coordinates.
2. Inclusion - struct field which gives the information about whether the nodule is cancerous or not. This is the prediction label for our dataset.

Below is the output snippet of the data frame generated:



This final Data Frame that is retrieved from the XML files has the structural schema as follows:

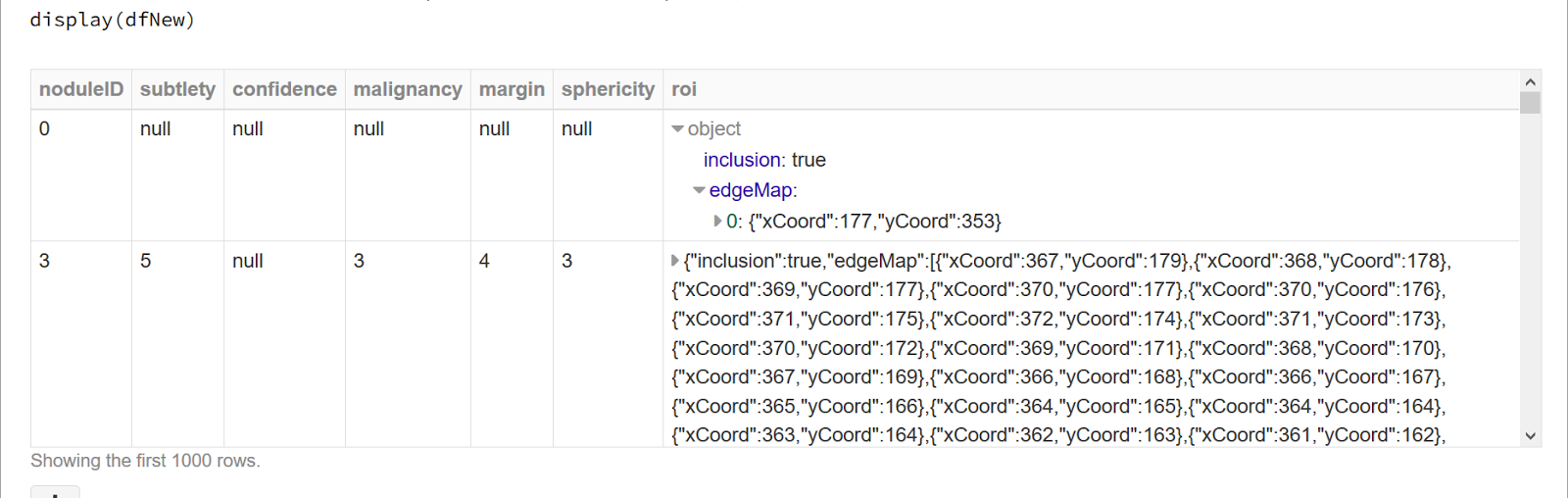


Now from the x and y coordinates we are forming a boundary and then calculating the total area of the nodule by integrating the thin sheets of area that are formed by using the boundaries created. Each layer of area is obtained by applying the area of polygon formula [4]:

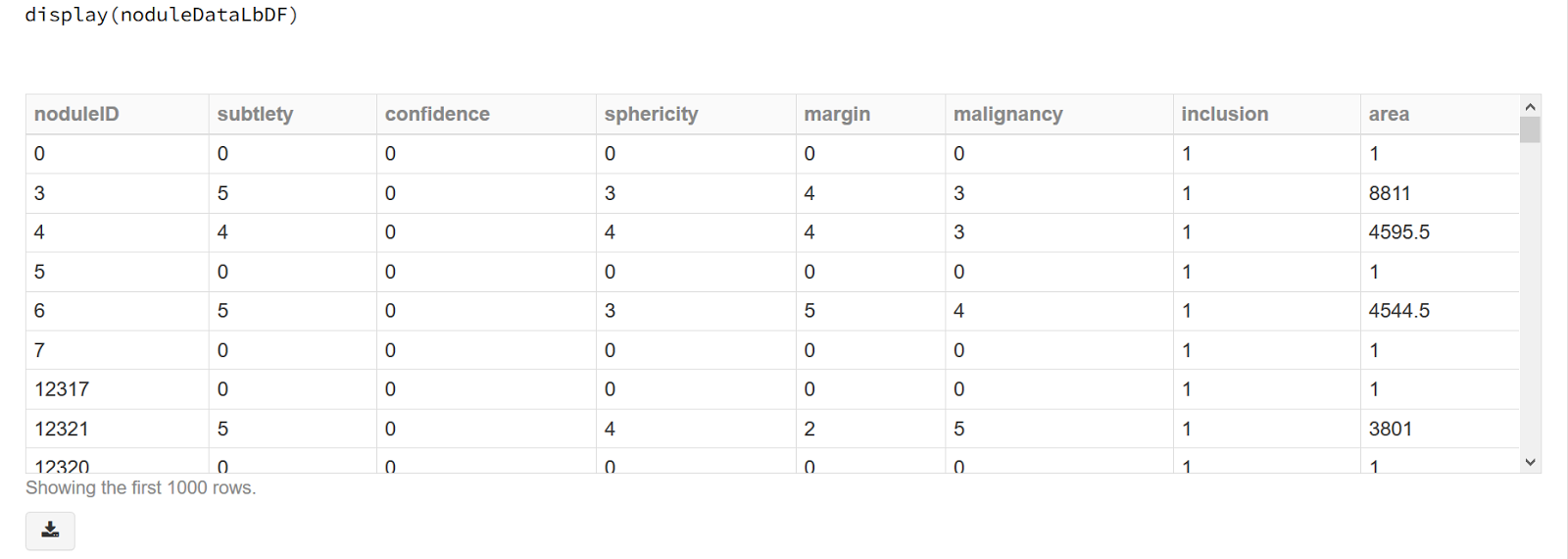
https://lh6.googleusercontent.com/hQi4s4TG1USIj_SF7mnPC_kFGxPvcHty9ABcx1y-UDhcGVVi3kLziMpkzWN_oHfNQ2Kf_Mz07cIc8bN2WoONpBB9fsM-I4ixBpy-opiUGj_lGXY9PIFp6e2HHAvT73LowgyJirpS

Using the explode function, all the columns which have multiple entries or multiple lists for a single row are converted into individual rows. This ensures that every column has a single entry for a particular row. The columns or tags on which the explode function is performed are unblindedReadNodule, unblindedRead, characteristics and roi.

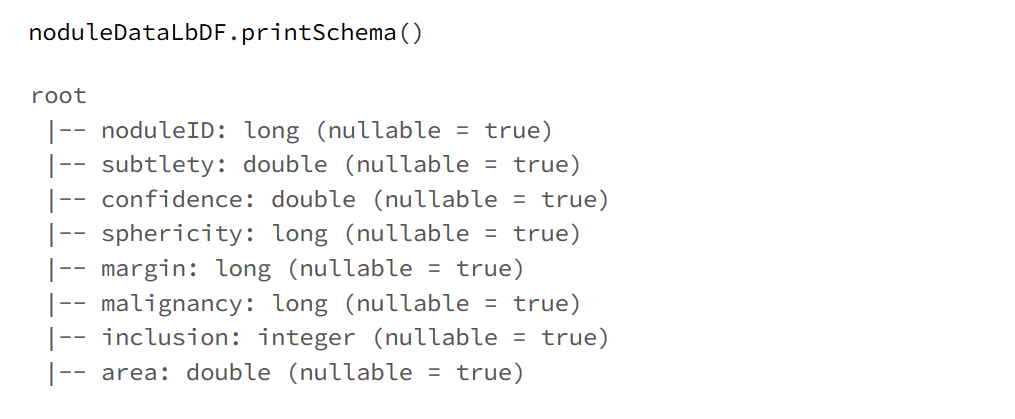
By cascading this explode function for the above mentioned columns, we obtain the output as shown below.



Now the resulting data frame from using the explode function is converted into rdd. We apply the map function to the obtained RDD to get all the columns and to add the area calculated for every noduleID respectively. Also, since we have a lot of null values in our data, we replaced them by 0 meaning the lowest level.



Below is how the final schema looks like after all the preprocessing:



# **PROPOSED SOLUTION AND METHODS**

We tried training using 2 models - SVM and Random Forest with various parameters and compared their results.

For this we used the following imports:

1. **RandomForestClassificationModel** : We used this library to build the random forest model with Max bins and Min instances per node.
2. **RandomForestClassifier** : We used this library in the Random forest model.
3. **BinaryClassEvaluator** : The Binary class evaluator is used as the evaluator.
4. **feature.{IndexToString, StringIndexer, VectorIndexer} :** This Libraries are used in to convert the label from string to index, and vectorindexer to for the transformation of the features.
5. **ml.tuning.{CrossValidator,ParamGridBuilder} :** This library function is used to perform the cross validation and build the param grid.
6. **Evaluation.RegressionEvaluator,.evaluation.MulticlassClassificationEvaluator :** This library function is used to perform the evaluation.
7. **ml.pipeline functions** : This ml library functions are used to build the pipeline.

We took the obtained preprocessed output from the previously performed preprocessing code and implemented the models in the scala.

From the obtained csv, we obtain a dataset by taking the inclusion as the label and rest of the features in a vector format.

Then we split the data obtained in to training data and testing data. We took 80% of the data as the training data and the remaining 20% as the testing data.

We considered two different models with various parameters:

1. **SVM Model:**

**Training Phase:**

We give this obtained training data as the input to the linear SVC model. We considered the regression parameter α to be 0.3.

The Pipeline is used for building the model. The evaluator is used for checking the labels with the correct ones. Here we considered Binary class evaluator for the evaluation of the data obtained from the train model. Evaluator is a metric to measure how well a fitted Model does on held-out test data. We checked the paramGrid for the maximum iterations of 10, 100, 150 and 200. The cross validation is performed by allocating the estimator, evaluator and paramGrid and number folds. This is done in five folds. In the final stage of training the obtained trained model is given as the input to the estimator.

**Testing Phase:**

The predictions for the testing phase are obtained from the output obtained by training a model. In the Testing phase the transformation operation is performed over the testing data. The obtained predictions are given as input to the evaluators to find the accuracy predictions. Here we use the binary class evaluator.

After getting the predictions from the testing phase, we obtained actual and predicted labels. We now built confusion matrix as shown below when used without PCA.

1. **Random Forest:**

We considered the Random Forest model for the obtained preprocessing data to compare the accuracy predictions with the previously performed linear SVM models to choose the most accurate model.

Here we divided the data by considering 80% of it as the training data and 20% of it as the testing data.

**Training Phase:**

Here we used RandomForestClassifier in pipeline for building the model. In parmGrid,

Max bins with the values 100, 500, 200, 50 and and minInstancesPerNode with values of 2, 5, 10 are considered as the parameters. Here we considered Binary class evaluator for the evaluation of the data obtained from the train model.The cross validation is performed by allocating the estimator, evaluator and paramGrid and number folds.This is done in five folds. In the final stage of training the obtained training data is given as the input to the estimator.

**Testing Phase:**

The predictions for the testing phase are obtained from the training phase output. In the Testing phase the transformation operation is performed over the testing data. The obtained predictions are given as input to the evaluators to find the accuracy of predictions.

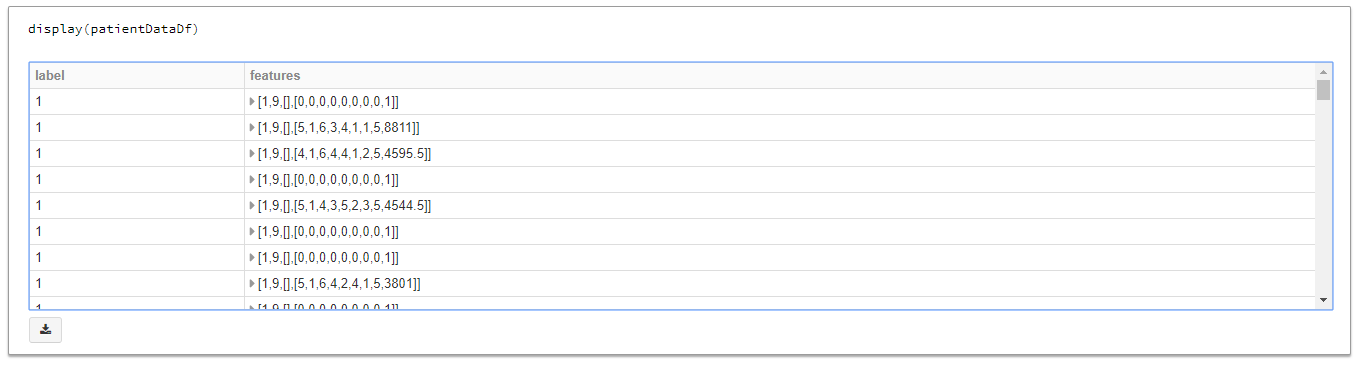
Here we plot the True positive, True negative, False positive and false negative over the original and predicted labels.

After this, we use the PCA model to consider the limited number of most relevant features. Here we set limit for the number of features as 6 and a 80- 20 split of the training data.

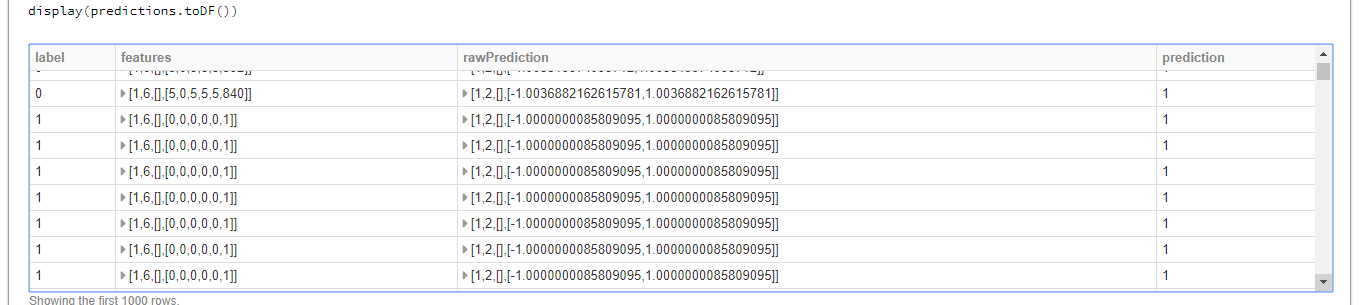
# **EXPERIMENTAL RESULTS AND ANALYSIS**

As the data is huge and there are multiple features that contribute to the prediction of the cancerous modules, we had to experiment several times with different parameters for each model. Below are some of the visual representations of the results of the analysis.

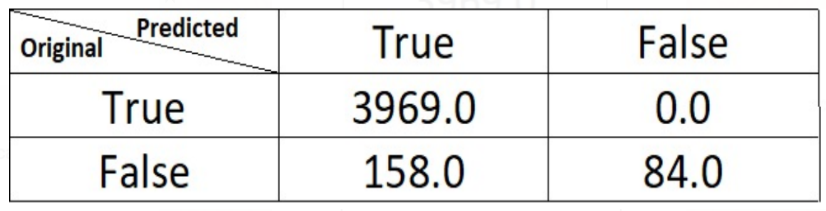
Vector format representation result of the data frame that is obtained from preprocessed data.



Predictions obtained from SVM Model:

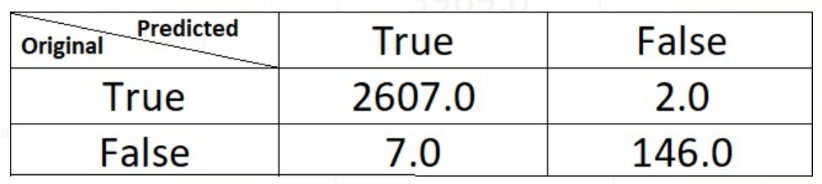


After getting the predictions from the testing phase, we obtained actual and predicted labels. We now built confusion matrix as shown below when used without PCA. Below are results obtained from SVC without PCA (which is not very good):



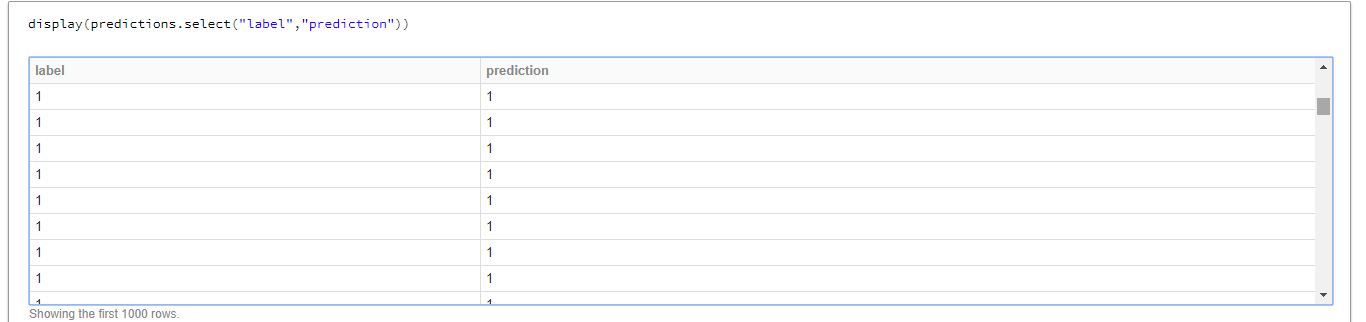


For Random Forest, we obtained the following confusion matrix and results:





And the following is the actual and predicted value outputs obtained:



Such experimental results are conducted and the following table is been prepared for each parameter of the SVM or Random Forest model and the results have been summarized in the following table:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Models | Split | Parameters | | | Accuracy (%) |
| SVM | 70:30 | Regression Parameters | | 0.3 | 96.25 |
| 0.2 | 99.74 |
| PCA with K values | | 5 | 95.23 |
| 6 | 99.72 |
| 80:20 | Regression Parameters | | 0.3 | 99.74 |
| 0.2 | 99.74 |
| PCA with K values | | 5 | 95.37 |
| 6 | 99.75 |
| RF | 70:30 | Paramgrid | Maxbins | 100, 500,200,50 | 99.76 |
| MinInstances per node | 2,5,10 |
| PCA with K values | | 5 | 99.74 |
| 6 | 99.74 |
| 80:20 | Paramgrid | Maxbins | 100,500,200,50 | 99.67 |
| MinInstances per node | 2,5,10 |
| PCA with K values | | 5 | 99.75 |
| 6 | 99.78 |

# **CONCLUSION**

Lung cancer is a leading condition in the world of cancer to cause deaths all over the world. This is majorly because of the fact that it can originate and spread with a lot of contributing factors like tobacco, smoke, pollution and other such factors. The reason of its notoriety being its difficulty of getting detected. To deal with this, we have built a model that would not only detect the cancer but with an added benefit of early detection.

With the different characteristics defined by Radiologist and doctors, we were able to define and build a machine learning model which when used will effectively detect early stages of cancer and thus help in their early treatment. We chose to implement a Random Forest model with PCA and k value 6.

# **CONTRIBUTIONS**

Preprocessing – Defining custom schemas, obtaining the final data frame through intermediate processing by elimination of outliers and empty values. Performing mathematical calculations like area based on the coordinate values, exploding the compressed dataframe to obtain uncompressed individual columns are done by all of the team members combined.

Model creation and experimental analysis is done using 2 techniques,

SVM – done by Mohana Priya and Lavanya Swetha

Random Forest – Sangeeta and Gautam

# **REFERENCES**

[1] <http://ieeexplore.ieee.org/document/6391662/>

[2] <http://cs231n.stanford.edu/reports/2017/pdfs/515.pdf>

[3] <http://ieeexplore.ieee.org/document/6391662/>

[4] <https://www.mathopenref.com/coordpolygonarea.html>