ME 4111 SCML Project

Classification of tumor types using LSVC and SVC with RBF kernel

Vivek Karmarkar 20 April 2021

# **Introduction:**

The Breast Cancer Wisconsin (Diagnostic) dataset has been used in this project. The data present in this dataset was obtained from the University of Wisconsin Hospitals, Madison. The data is a labelled data set, where the label corresponds to the class of the breast cancer tumor. The class of the tumor is either benign or malignant. The dataset has 10 features related to the tumor and a total of 699 instances.

The related problem is that of classifying the tumor as benign or malignant based on 2 or more features of the tumor using the LSVC and SVC with RBF kernel approach.

The dataset was obtained from the following link: <https://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+(Diagnostic)>

The name of the specific dataset used:

breast-cancer-wisconsin.data and breast-cancer-wisconsin.names

# **Methodology:**

Since this is a classification problem, the LSVC and SVC with RBF kernel algorithms were used.

1. **LSVC:** The Linear Support Vector Classifier is a classification algorithm based on a linear decision boundary as the name suggests. The LSVC algorithm helps classify data as belonging to different classes based on their feature values. In 2D, the decision boundary is a straight line. In 3D, the decision boundary is a plane and in higher dimensions, it is a hyperplane. The LSVC hyperplane is chosen to maximize the “margin” or the distance from data points of different classes. The corresponding hyperparameter associated with the LSVC is called “C”. The “C” hyperparameter is related to the amount of misclassification. For larger values of “C”, small margins are chosen to avoid misclassification whereas lower values of “C” may lead to misclassification. The default value used for C=1.
2. **SVC with RBF kernel:** Data is not always linearly separable and, in such cases, the LSVC is not very useful since the decision boundary will be non-linear. For these set of problems, a kernel trick is used to transform the data into a higher dimensional space where it is linearly separable. The SVC with RBF kernel algorithm belongs to this class and is used to classify data sets where the classes have a non-linear decision boundary. The hyperparameters associated with this method are called “C” and “gamma”.

# **Data Exploring:**

## **Exploratory Data Analysis:**

The data was read into a pandas dataframe and the head and tail commands were used to visualize the first and last 5 entries of the dataset in a tabular format. It was clear that the class variable feature that corresponded to benign and malignant should be temporarily transformed from ‘2’ and ‘4’ to ‘B’ and ‘M’ for the purposes of the exploratory data analysis part to simplify things. We also wanted to make sure that they were only 2 unique classes and that the dimensions of the dataframe were 699x11. We proceeded with checking the number of unique classes, the dimensions of the dataframe and carrying out the temporary transformation of the class variable. Following this, a plot was made (Figure 1 below) to visualize the number of benign and malignant tumors.

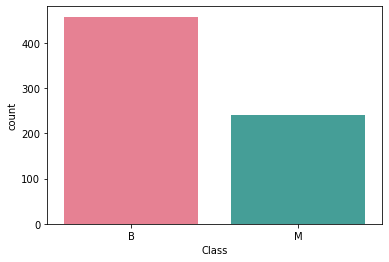


Figure : Visualizing the number of Benign and Malignant tumors

The other thing that was clear was that the feature called “Sample code number” was not relevant and hence, could be removed without affecting the rest of the analysis. Before removing this specific feature, the dataset was checked for other features that may be removed for the sake of simplifying the analysis. The dtypes command was used to check the various datatypes that the features corresponded to. It was found that all features except the “Bare Nuclei” feature were int64 types having values between 1 and 10. It was concluded that the “Sample code number” and “Bare Nuclei” feature should be removed for the purposes of simplifying the analysis. The drop command was used to remove the features. Following this, the head, tail and describe commands were used to make sure that the dataframe appeared as expected.

## **Select two or more features for data analysis:**

Since it was decided that a classification algorithm was to be used, the next step was to decide if two or more features were to be used and how these features would be decided upon. It was decided to use 2 features since 2 features can be easily visualized and this helps in validating results with graphs and plots. The next step was to construct a pair-pair plot using the seaborn library in order to visualize the features which had best spatial separation.

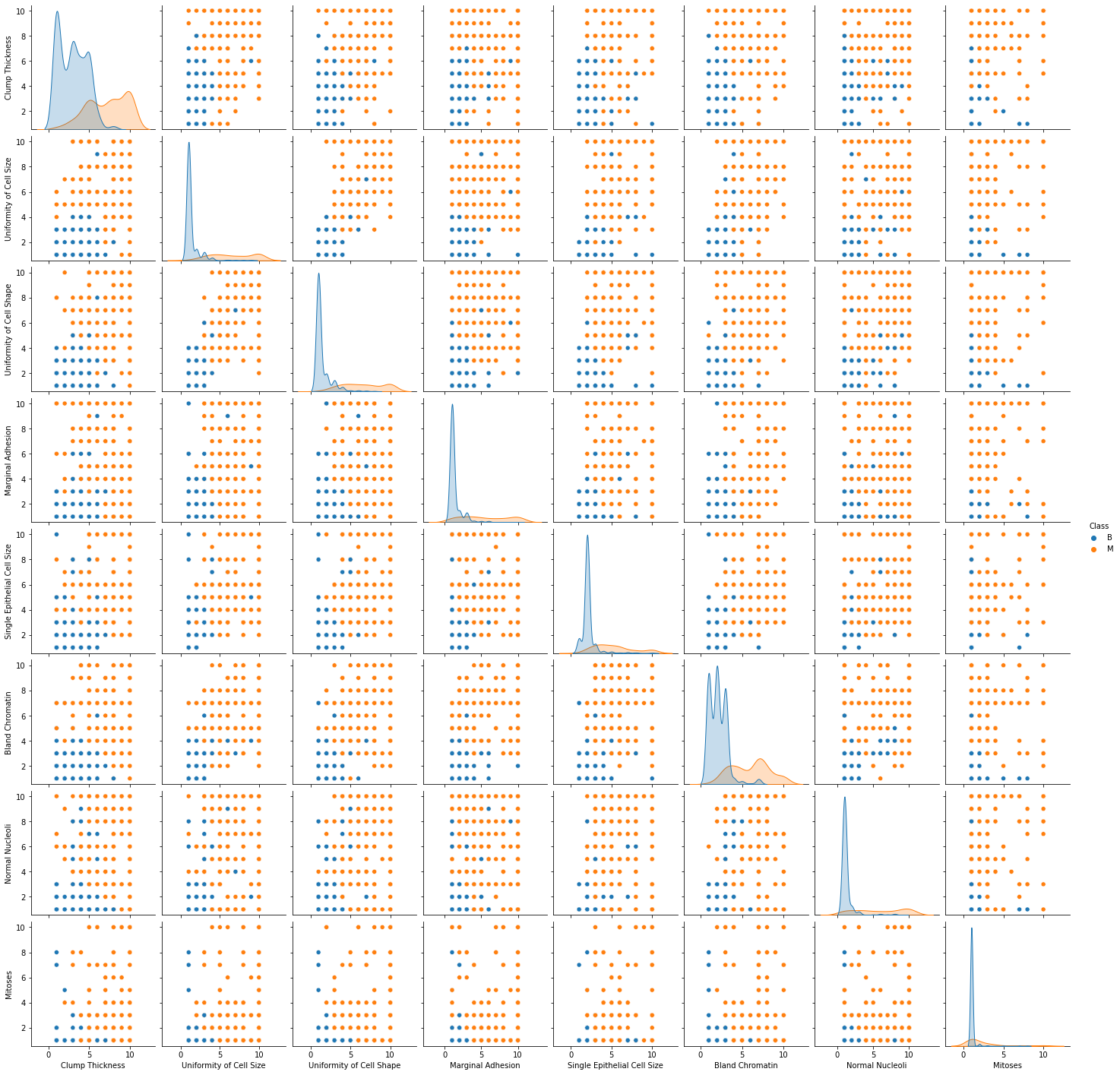


Figure : Pair-Pair plot of the features

As can be seen from the above plots (Figure 2 above), the plots on the diagonal give a good indication of which features may have the best spatial separation for benign and malignant tumors. It appears as if the orange and blue portions of the plots on the diagonal for the features corresponding to “Uniformity of Cell Size” and “Uniformity of Cell Shape” have least overlap. The off-diagonal plots located at (row 2, column 3) and (row 3, column 2) also show good separation between these two features. Violin plots, swarm plots, distplots and histograms were used for further visualization of just these two features.

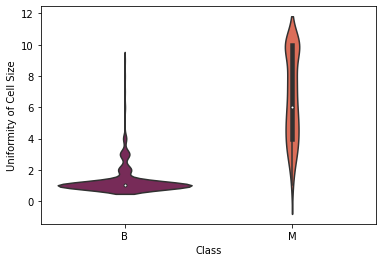


Figure : Violin Plot for Uniformity of Cell Size

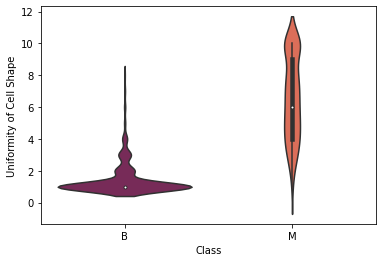


Figure : Violin Plot for Uniformity of Cell Shape

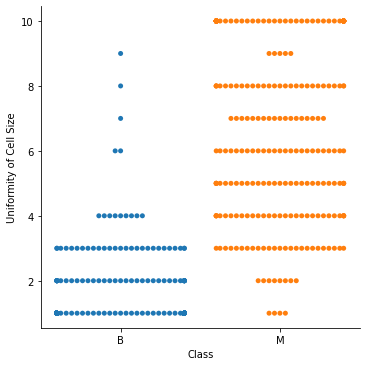


Figure : Swarm Plot for Uniformity of Cell Size

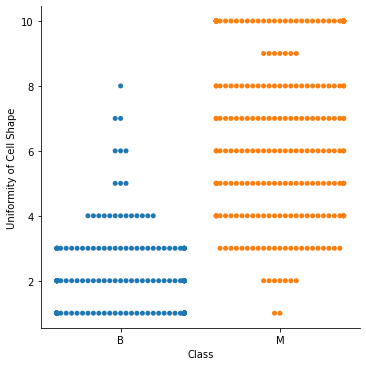


Figure : Swarm Plot for Uniformity of Cell Shape

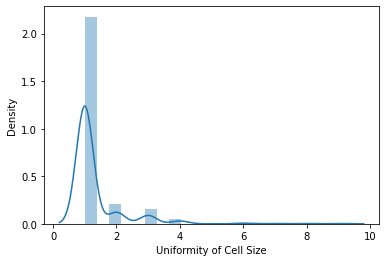


Figure : Distplot for Benign Tumor (Uniformity of Cell Size)

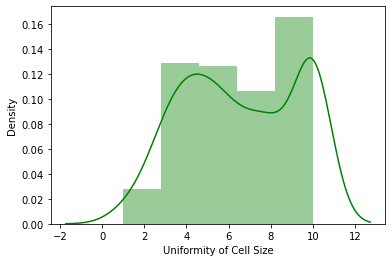


Figure : Distplot for Malignant Tumor (Uniformity of Cell Size)

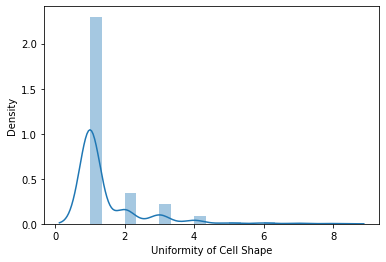


Figure : Distplot for Benign Tumor (Uniformity of Cell Shape)

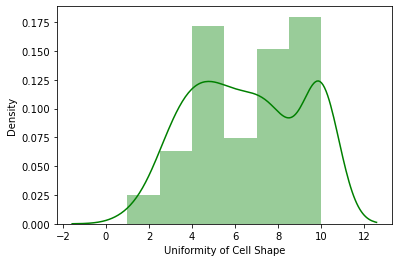


Figure : Distplot for Malignant Tumor (Uniformity of Cell Shape)

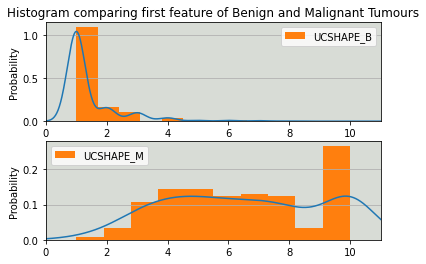


Figure : Histogram comparing Uniformity of Cell Shape (Above: Benign and Below: Malignant)

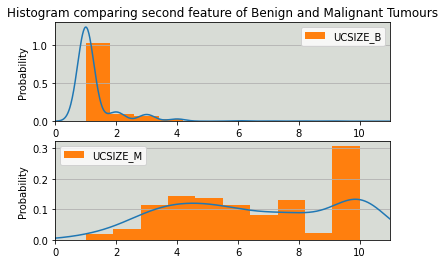


Figure : Histogram comparing Uniformity of Cell Size (Above: Benign and Below: Malignant)

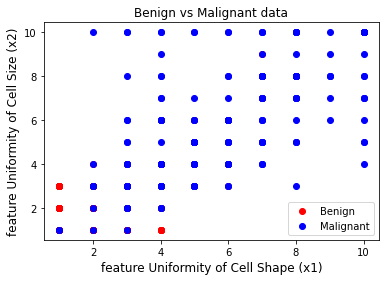


Figure : Scatter plot for Uniformity of Cell Size and Uniformity of Cell Shape

Based on all the above plots, it appeared as if the two features namely, Uniformity of Cell Size and Uniformity of Cell Shape may be useable as the two features for our classification scheme. It was decided to perform a final analysis to gauge spatial separation using a table displaying the difference between the mean values of various features for benign and malignant tumors as well as the standard deviation of those features. Before generating the table, the class variables were transformed back to numeric variables. The dtypes, describe, head and tail commands were used to make sure the data appeared as expected. Finally, the isnull.sum command was used to check if any null values were present. No null values were found, so we proceeded with splitting the dataset based on class and generating the table described earlier in this paragraph.

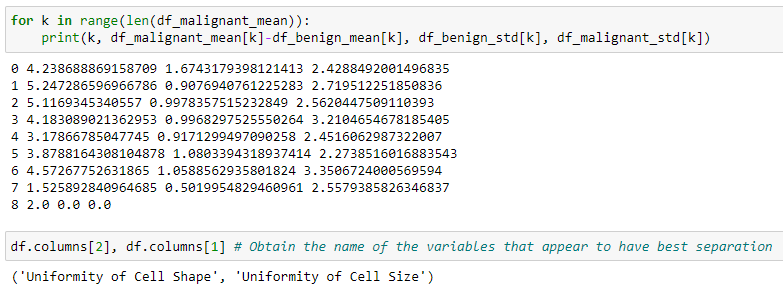


Figure : Table to visualize spatial separation between features

As we can see from the table above (Figure 14), the indices of the rows that correspond to variables having the best spatial separation are k=1 and k=2 and the corresponding feature names are “Uniformity of Cell Size” and “Uniformity of Cell Shape”. The means have the highest separation and the standard deviation values are also on the lower end indicating good spatial separation. This agrees with all the different plots that we investigated earlier. Based on this analysis, it was decided that the two features to be used were “Uniformity of Cell Size” and “Uniformity of Cell Shape”.

# **Machine Learning:**

This part was started by splitting the dataframe into a features array ‘X’ and a labels array ‘y’. The ‘X’ array contained the data corresponding to the two selected features, namely ‘Uniformity of Cell Shape’ and ‘Uniformity of Cell Size’. The ‘y’ array was converted into a binary array based on condition y==2 for benign tumors. The data was split twice with a random state=2 to obtain the training, validation and test datasets. The standard scaler object was imported and was used to scale the training, validation and test datasets to avoid ill-conditioning. The test dataset comprised 25% of the total dataset. A plot showing the training data after scaling is shown below (Figure 15 below).

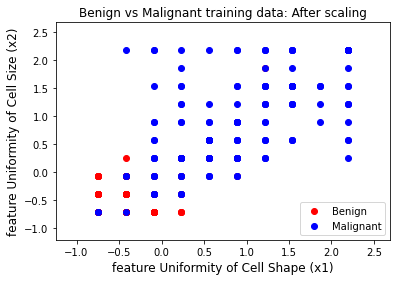


Figure : Plot of training data after scaling

Following the splitting and scaling of the dataset, we proceeded to fit the LSVC to the training data for a range of values for the hyperparameter “C” and evaluated a given fit using the validation data. A table displaying the values (Figure 16 below) of “C”, training accuracy, validation accuracy as well as the recall, precision, F1 scores for the validation data was constructed. It was observed that the performance of the algorithm on the validation data did not vary much as “C” was changed and therefore the default value of C=1 was chosen.

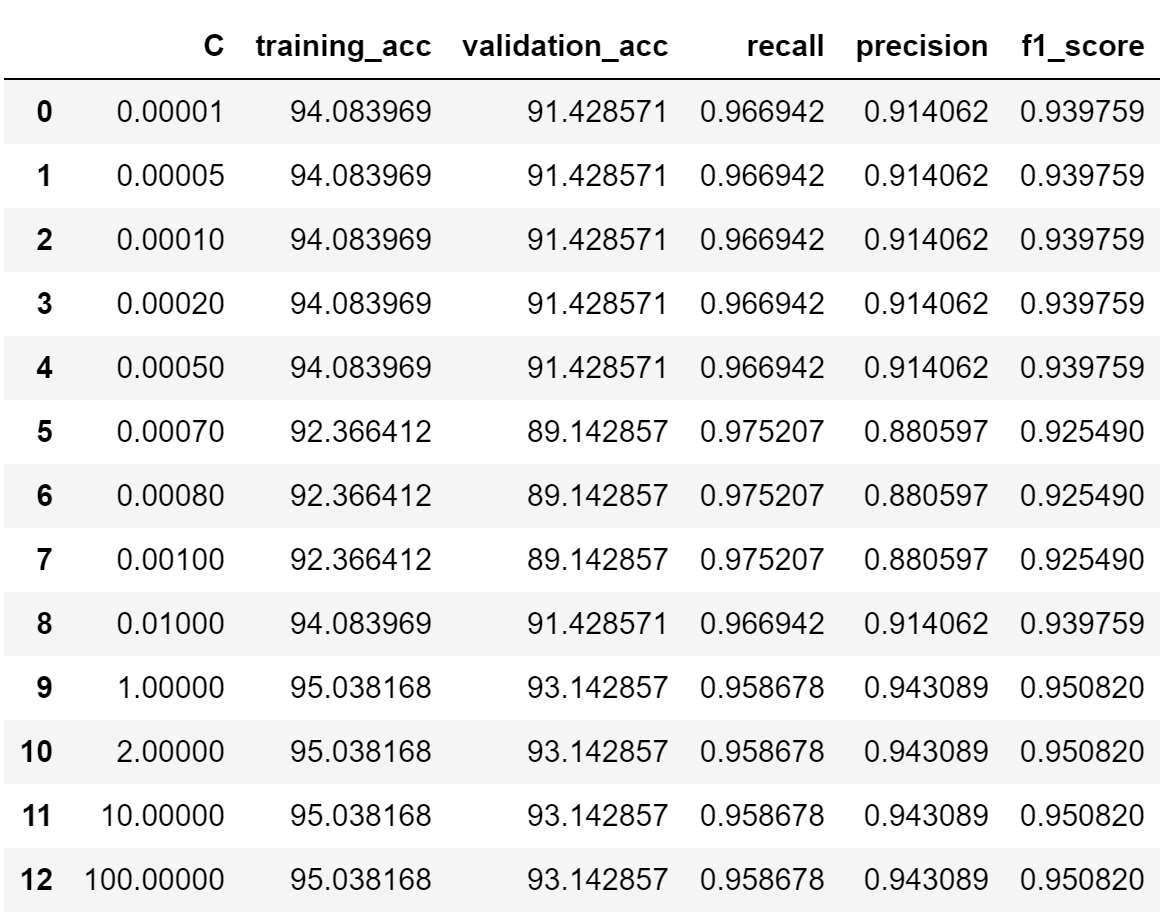


Figure : Table for selection of "C"

A similar table for the SVC with RBF kernel for the hyperparameter “gamma” was constructed where “gamma” was varied from 0.00001 to 1.0 for a given value of “C”. The C hyperparameter was varied from 0.001 to 100.0 in non-uniform increments. It was seen that for values of C less than 0.5, underfitting was obtained for all values of gamma whereas for value of C greater than 50, overfitting was obtained for all values of gamma. For intermediate values of C, The training and validation accuracy were seen to increase as gamma was increased and was greater than 90% for values of gamma greater than some critical gamma indicating that that the algorithm may be overfitting at these values of gamma. Amongst the different values of C investigated, the default value of C=1 seemed to have the least amount of overfitting as well as the least amount of underfitting (see Python code for more details) as gamma was varied. Therefore C=1 was chosen. For C=1, gamma = 0.0006 had training accuracy and validation accuracy values around 85% and this seemed like a good choice. Based on the above analysis, C=1 was used for the LSVC and (C=1, gamma=0.0006) was used the SVC with RBF kernel. The table is shown below (Figure 17 below).

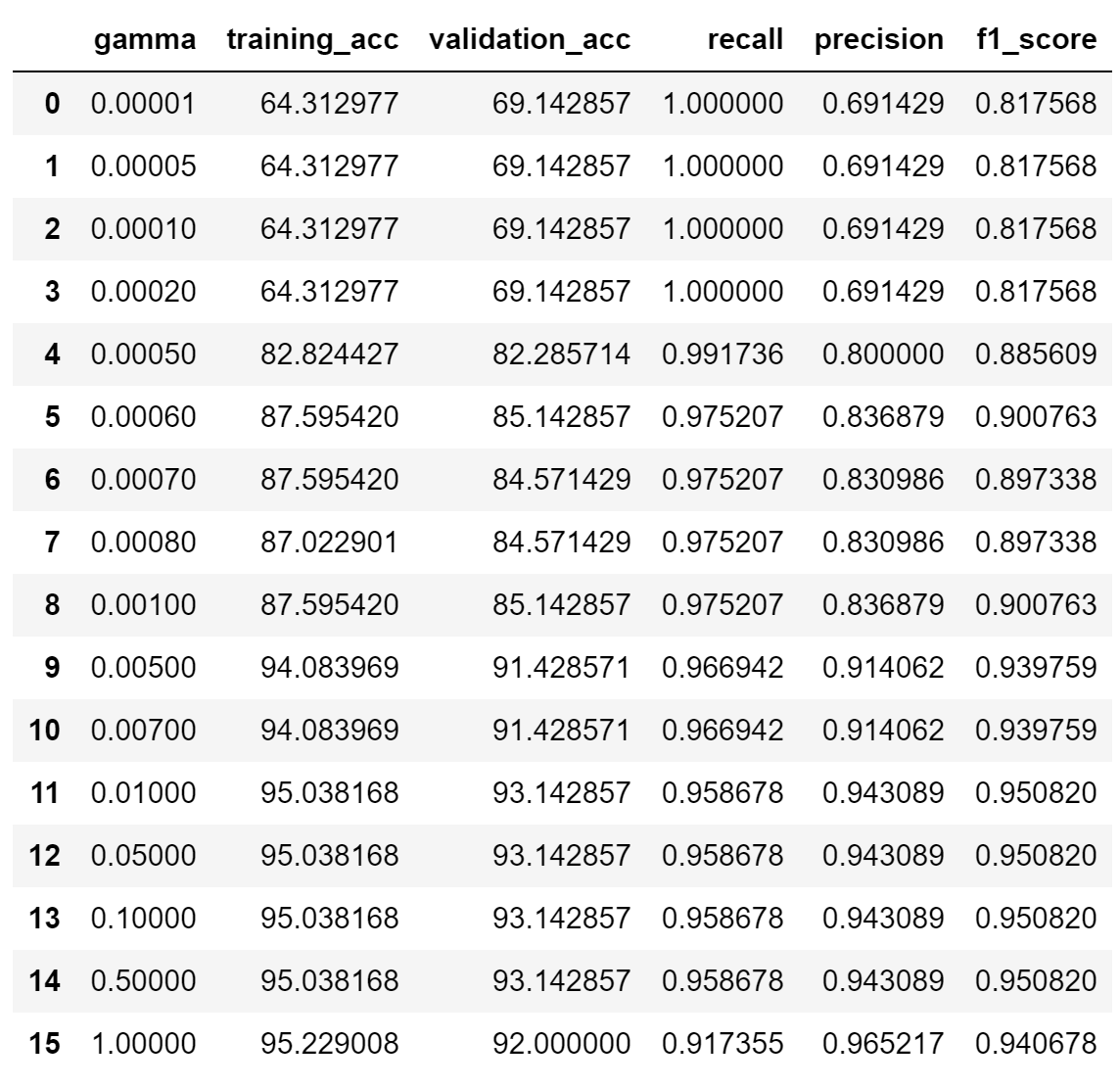


Figure : Table for selection of gamma

Now that the hyperparameter tuning has been achieved based on fitting the data to the training set and then evaluating the performance on the validation set, we can look at plots comparing the decision boundaries with respect to the training and test data as well as confusion matrix reports and heatmaps for the test data set.

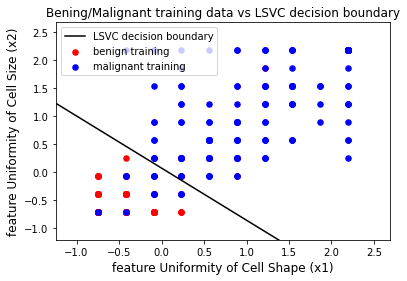


Figure : LSVC decision boundary with training data

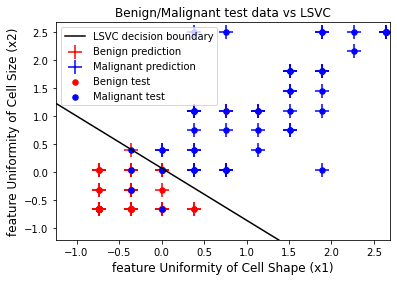


Figure : LSVC decision boundary with test data

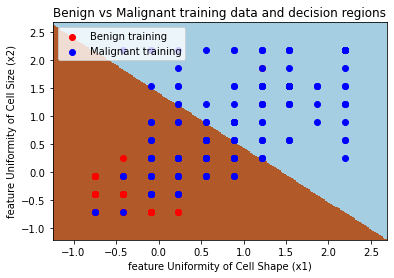


Figure : SVC with RBF kernel decision boundary with training data

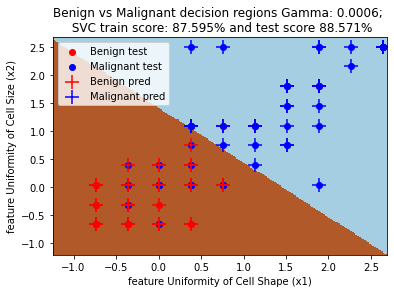


Figure : SVC with RBF kernel decision boundary with test data

## **Confusion Matrix for LSVC below:**

[ 47 7

5 116]

Precision for LSVC: 0.943

Recall for LSVC: 0.959

F1 score for LSVC: 0.951

Accuracy of LSVC classifier on training set: 95.04%

Accuracy of LSVC classifier on validation set: 93.14%

Accuracy of LSVC classifier on test set: 93.14%

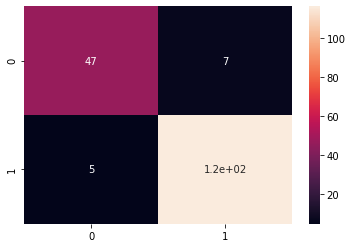


Figure : Heatmap for Confusion Matrix of LSVC

## **Confusion Matrix for SVC with RBF kernel below:**

[ 37 17

3 118]

Precision for SVC with RBF kernel: 0.874

Recall for SVC with RBF kernel: 0.975

F1 score for SVC with RBF kernel: 0.922

Accuracy of SVF with RBF kernel classifier on training set: 87.60%

Accuracy of SVC with RBF classifier on validation set: 85.14%

Accuracy of SVC with RBF kernel classifier on test set: 88.57%

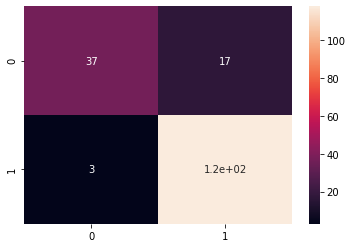


Figure : Heatmap for Confusion Matrix of SVC with RBF kernel

# **Discussion and conclusion:**

A LSVC (C=1.0) and SVC with RBF kernel (C=1.0, gamma=0.0006) were used to classify the tumor as benign or malignant. The models were fit to the training data and evaluated on validation data for a various range of hyperparameters before deciding on the final value of the hyperparameters to be used. After tuning the hyperparameters, these models were evaluated on the test data which comprises 25% of the size of the total dataset.

The plots for the LSVC showing the linear decision boundary indicate good performance on both the training and test data. This is also reflected in the accuracy values for the test, validation and training data sets being greater than 90% as well as in the precision, recall and F1 scores being greater than 0.94 for the test data.

The plots for the SVC with RBF kernel also have a linear decision boundary. The lack of the non-linear decision boundary may be based on the nature of the spatial arrangement of the data. It appears to be linearly separable and therefore, a nonlinear decision boundary does not apply here. However, the plots for the SVC with the RBF kernel seem to indicate a poorer performance as compared to the LSVC plots. This is since the decision boundary lines corresponding to the SVC with the RBF kernel have greater axes intercept values thereby leading to heavier misclassification of malignant tumors as compared to the LSVC. But this very phenomenon also leads to a conservative classification of the benign tumors. These observations are reflected by the fact that the accuracy values as well as the precision and F1 scores are lower than the LSVC but the recall score being higher than the LSVC.

Overall, the LSVC seems to do a better job than the SVC with RBF kernel at classifying benign and malignant tumors based on two selected features namely, Uniformity of Cell Size and Uniformity of Cell Shape. We also assumed that these features had good spatial separation based on some preliminary data exploration. This seems to be the case based on the performance of the classification algorithms used. Another observation which was not discovered during the preliminary data exploration is that the two selected features are linearly separable.

# **Each Member’s contribution:**

This project was completely done by one person, me.