**ML Project**

Muhammad Haris Sohail

Zeeshan Manzoor

Faizan Zahid

Department of Economics, Information Technology University

Machine Learning for Economics

Sir Basit Ali

7, January 2022

**Introduction**

Question: “Predict whether the tumor is benign or malignant".

Breast cancer is a malignant cell growth in the breast. If left untreated, cancer spreads to other areas of the body. The difference between the two types of tumors, non-cancerous and cancerous, is important. A benign tumor is non-cancerous and not dangerous on its own. But a malignant tumor means the mass is cancerous.

The problem addressed in this paper is to develop a classifier that can best predict whether the tumor is benign or malignant. It's a classification model. The most important is to confirm if the tumor is malignant. This can be used to efficiently treat the tumor in a timely manner.

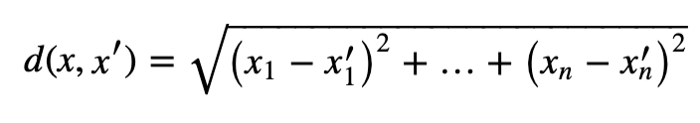
**Literature Review**

Machine learning has different types of learnings, and for each, different algorithms can be used. Our domain is restricted to supervised learning, which is the type of learning that includes class labels (we know them beforehand). One of the most common and effective algorithms is the K-Nearest-Neighbors or KNN. It is a classification method. Classification tasks are the most commonly encountered data mining tasks. These tasks, as described earlier, involve mapping an object to one of a set of predefined classes (Zhang et al., 2016). kNN is a case-based learning method which keeps all the training data for classification. As kNN is a simple but effective method for classification, it motivates us to build a model for kNN to improve its efficiency whilst preserving its classification accuracy as well. It's distance-based learning where the data is disturbed and divided in the same specific classes.

During the model construction process, each data point has its largest local neighborhood which covers the best number of data points with the same class label. Based on these local neighborhoods, the largest local neighborhood can be obtained in each cycle (Duo et al., 2003). This largest global neighborhood can be seen as a representative to represent all the data points covered by it. For data points not covered by any representatives, we repeat the above operation until all the data points have been covered by chosen representatives.

The number of cycles, represented by k, depends from model to model. kNN algorithm computes the distance between each training sample and test samples in the dataset and then returns k closest samples (Zhang, 2018). This is a classic classifier algorithm used for classifier problems, just like the one we have in our project.

For distance metrics, we use the Euclidean metric.

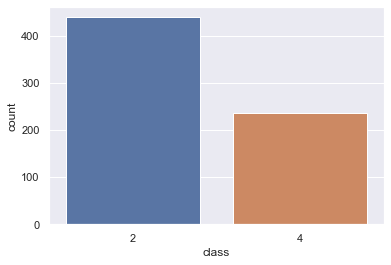


The input x gets assigned to the class with the largest probability.

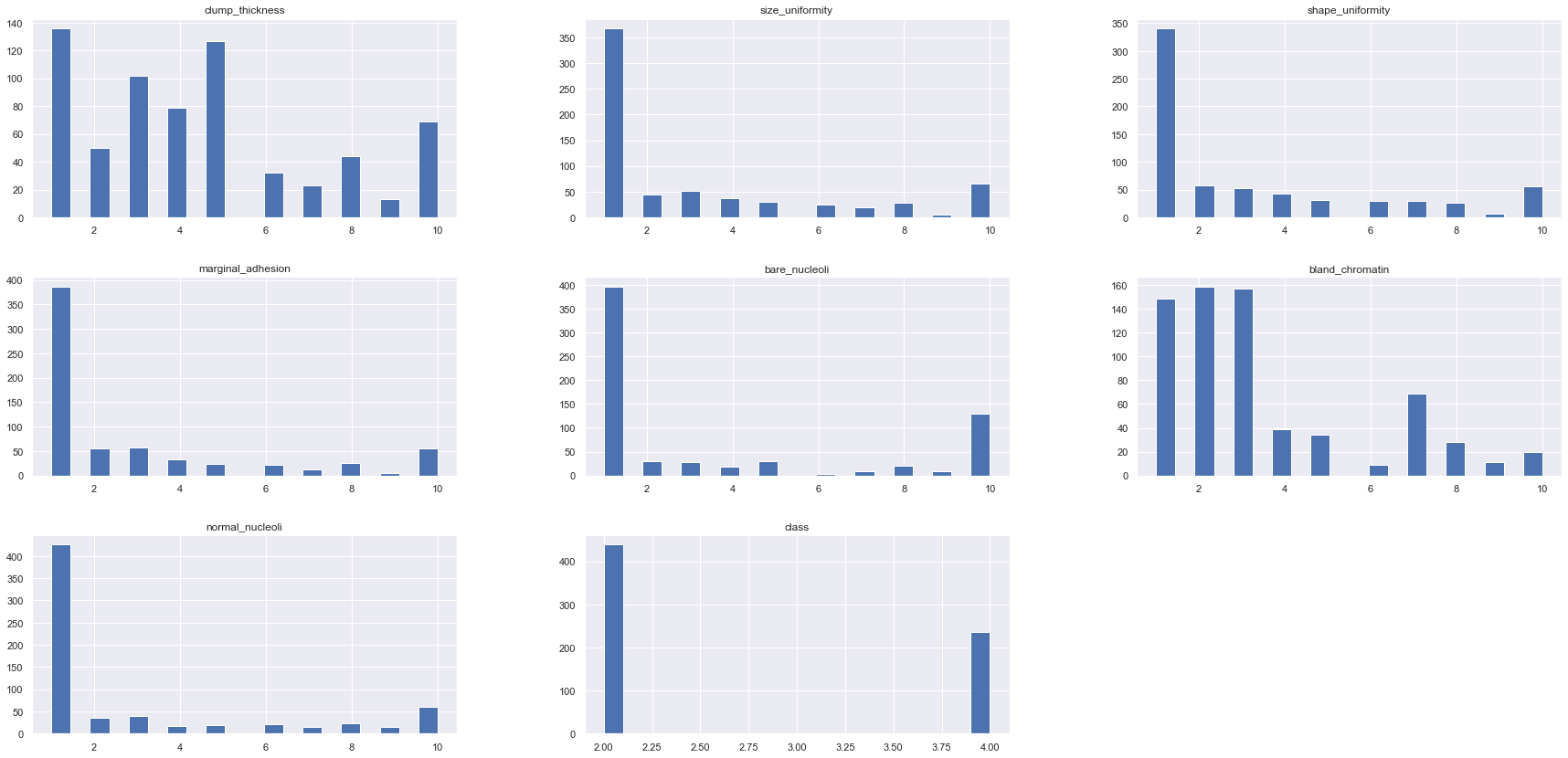
KNN is a very common and effective algorithm as it can be used for both classification and regression problems. In classification, the values are discrete, like in our model. For Regression, the technique will be the same; just instead of the classes of the neighbors, we will take the value of the target and find the target value for the unseen datapoint by taking an average, mean, or any suitable function you want.

**EDA**

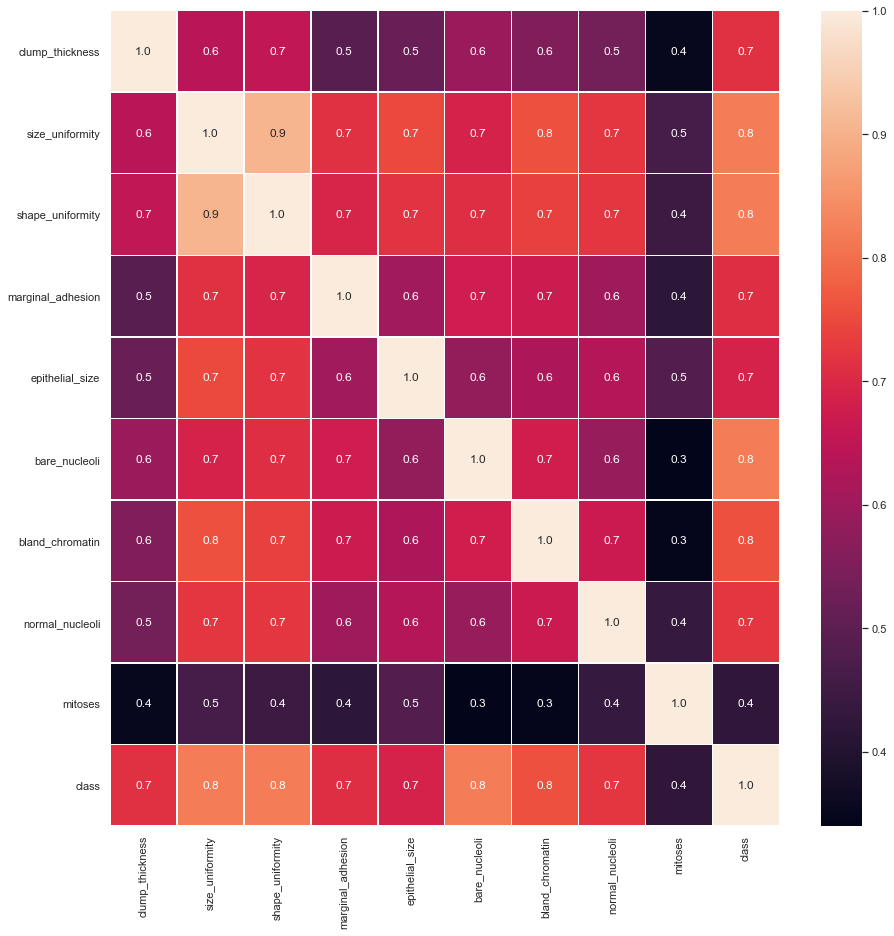
We were given breast cancer dataset of patients. The dataset was somewhat clean as it did not have any null values. But there were missing values in that had to be taken care of using some cleaning techniques. They were in ‘?’ form. Because the values were very less (just 16), they were dropped from the dataset. After that, we normalized and checked for duplicates and started the pre-processing and EDA. “Class” column is the target variable and it has two values, 2 for benign (non-cancerous), 4 for malignant (cancerous).



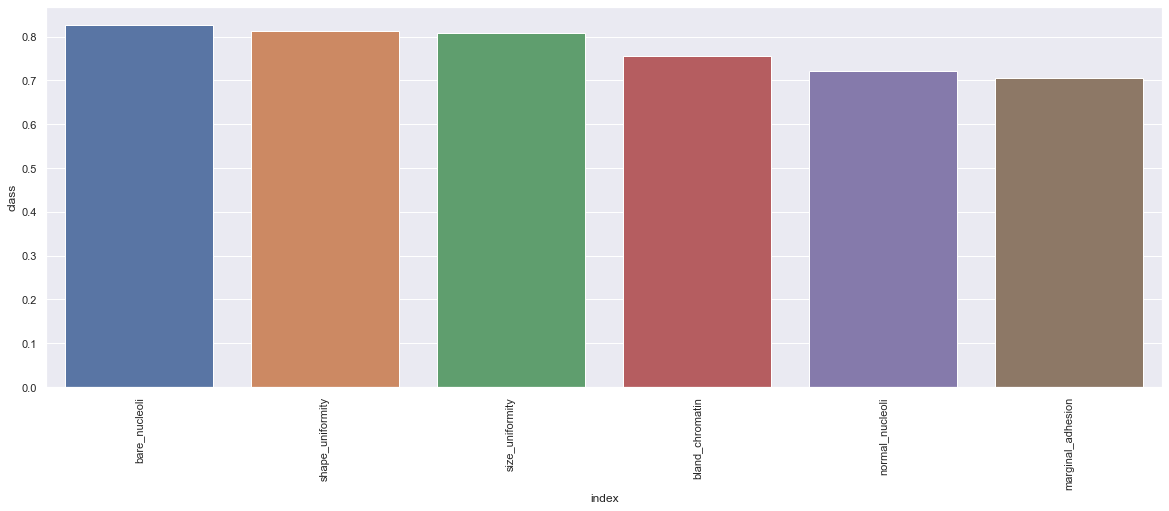
This graph shows us the distribution of the two types of tumors. It’s uneven but expected because benign tumor is more common than malignant. Next, in the program, we have shown the different types of distributions of the other attributes of compared to the each other which gives us a pretty clear correlation, both high and low. Intuition is important here because we can view the impact of different attributes on the cancer cells. This is one graph, a better distribution of each attribute is shown properly in the program file.



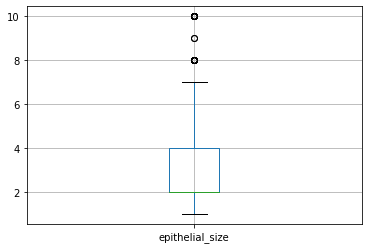
We have also shown a scatter plot. Another graph has been plotted too that gives the heatmap of the whole dataset, after dropping the ID column. It shows us the correlation between all the attributes and how they are affecting the target column, which is “Class”.

****

A bar plot shows the correlation of different attributes. This one is different because it only includes attributes with only high correlation; greater than 70 percent. This is an important step in Principal Component Analysis, to select the features that have the highest correlation. We have done pre-processing to include features with the highest correlation in one of the three models.



As for outliers, we have shown outliers of all the attributes in separate graphs. This was an important set as the outliers skewed the model. By removing them, we were able to achieve 99.2 percent accuracy. One is shown here, others are present in the program.



**Methodology**

The machine learning model we have used in our project is classification, which comes under supervised learning. Our dataset was labeled with discrete values, which is why we went with this model and also because we had studied and practiced it in the course. We have used two algorithms in our model: one is the K-Nearest Neighbors, and the other being Gaussian-Naïve Bayes (GaussianNB). We have already explained kNN, as for this other one; Naive Bayes are a group of supervised machine learning classification algorithms based on the Bayes theorem. It is a simple classification technique but has high functionality. It is a special type of gaussian algorithm used when features are following a normal distribution (Kamel, et al., 2019). We chose two algorithms to compare which best fits our dataset. GaussianNB works with continuous data.

The hyperparameter in kNN algorithm was the number of neighbors that were to be predefined. We performed two tests with this algorithm and kept the value of k different in both.

**Evaluation Process**

*Accuracy:*

As we used two algorithms, we will be comparing the accuracy scores of both. We got high accuracies with both algorithms. We ran two tests with the kNN algorithm, both with different methods. In the second test, instead of removing outliers, we tried to reduce the dimensionality but selecting features with high correlation and dropping others. But that didn't give a higher accuracy (it came out to be 96%) than the first test in which we just removed the outliers. This means the dropped features, even though they were of low correlation, were important to the model.

The highest accuracy from kNN came out to be 99.2 %. Results from both test runs are shown in the project. When the value of 'K' was at low, it gave lower accuracy, of 98.5 %, compared to when the value assigned was '5' (99.2%). This is because when K is small, we are holding the region of a given prediction and pushing our classifier to be "more blind" to the overall distribution. A small value for K provides the most adjustable fit, which will have low bias but high variance.

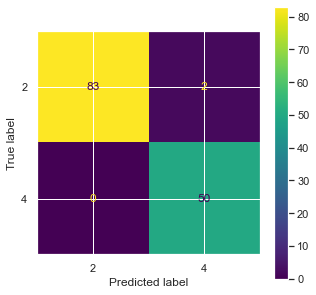
As for our GaussianNB algorithm gave us an accuracy score of 96%, which is quite high too but not higher than the one we got from kNN. Hence, we chose kNN as the final algorithm to go with.

**Highest achieved accuracy: 99.2 percent**

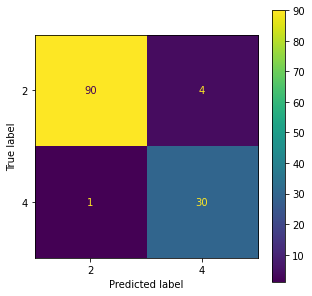
*Confusion Matrix*

A confusion matrix is a performance measurement method for Machine learning classification. It helps us to know the performance of the classification model on a set of test data for that the true values and false are known. It helps us determine how many times our model has given correct or wrong output and what type. Hence, it is a very important tool for evaluating classification models. It gives insight not only into the errors being made by your classifier but, more importantly, the types of errors that are being made. Confusion matrices were computed for both our ML model and also shown graphically.

For our first kNN model, we can see that kNN's predictions were quite good. Model made correct predictions in the case of True Positive, meaning 50 test subjects were correctly shown to have cancer while 90 correctly didn't have. In 4 cases, the model predicted True for something that wasn't correct; False Positive. As for False Negative, 0 mistakes was made.



As for the GaussianNB model, the model predicted almost similar results with True Positives (30) and True Negatives (90). It predicted just one wrong False Negative, predicting No when the test had cancer. As for False Positive, the model made mistakes with the same number of test subjects, i.e., four.



**Conclusion**

This was our project on predicting whether a tumor in patient is benign or malignant. We implemented machine learning models taught to us during the course and achieved high accuracy in our models. The practical approach of this course has helped us in model building process of machine learning, which is very important. Even though the data given to us was somewhat clean, it still had to be processed and cleaned further for proper use. The models we implemented showed high accuracy in predicting, and in the end confusion matrices were also helpful in showing the performance of the models. Out of the three models, (two kNN and one GaussianNB) we chose kNN because of the high accuracy, at 99.2 percent. This is a very good accuracy score for our first ever model.

**References**

Kamel, H., et al. (2019). Cancer Classification Using Gaussian Naive Bayes Algorithm. 2019 International Engineering Conference (IEC). doi:10.1109/iec47844.2019.8950650

Guo, G., et al. (2003). KNN Model-Based Approach in Classification. *Lecture Notes in Computer Science*, 986–996. doi:10.1007/978-3-540-39964-3\_62

Zhang, S., et al. (2016). Efficient k NN classification algorithm for big data. *Neurocomputing*, 195, 143–148. doi:10.1016/j.neucom.2015.08.112

Zhang, S., et al. (2018). kNN Classification With Different Numbers of Nearest Neighbors. *IEEE Transactions on Neural Networks and Learning Systems*, 29(5), 1774–1785. doi:10.1109/tnnls.2017.2673241