NB and LVQ Classifiers

In this assignment Wisconsin breast cancer data set was used to train a Naïve Bayes classifier and a linear vector Quantization classifiers using python's *sklearn.naive_bayes. GaussianNB* and *sklearn_lvq.GlvqModel* libraries. First the data set was loaded into a pandas data frame and observed to be consisting of 10 features and a class label representing whether each data point represents a 'Benign' or 'Malignant' breast cancer. Since original data set did not have feature names assigned in the columns, appropriate names were given to each of the 11 columns including the class label.

It was observed that data set contained few missing values on 'Bare_Nuclei' feature that was represented as '?' value. To cater this, initially the missing values were replaced with null (np.nan) and then python's univariate imputation library was used to replace the null values with the most frequent value of each feature. After imputation, data types of the columns were observed to have changed to type object, and therefore they were converted back to original data type that was int64. After handling missing values, the feature representing 'sample code number' was removed from the data set as it did not represent any real feature but a code number given to each sample. sklearn.model_selection.train_test_split was used to split the original data into training and test sets using the same ratio as done in earlier assignments that was 35%.

For Naïve Bayes classification, *sklearn.naive_bayes.GaussianNB* module was used leaving all the default parameters and above split train data set was used to train the model. Then the test data set was used with the trained model to predict the results which were then used to generate *classification_report* against the real test values as below.

	precision	recall	F1-score	support
Benign	0.99	0.95	0.97	150
Malignant	0.92	0.99	0.95	94

sklearn.metrics.confusion_matrix method was used to generate the confusion matrix for above results and the results are 93 true positives, 142 true negatives, 8 false positives, 1 false negatives. *sklearn.metrics.accuracy_score* method was used to calculate the overall accuracy of the model to be 96.3% and this was the second highest accuracy obtained for this data set so far when compared with C4.5 (90%), logistic regression (93%) , Random forest (95.9%) and KNN (97.5%).

For LVQ classification above same imputed data set without 'sample code number' feature (9 features in total) was trained (35% test-train ratio) with Generalized Learning Vector Quantization (GVLQ) model from *sklearn_lvq.GlvqModel*. After training the model with train data set, classification report was generated using predicted values of the test set and real values of the test set.

	precision	recall	F1-score	support
Benign	0.99	0.97	0.98	150
Malignant	0.95	0.98	0.96	94

Confusion matrix claimed a result of, 92 true positives, 145 true negatives, 5 false positives, 2 false negatives. Overall classification accuracy of the model was given as 97.13% resulting LVQ to become the model with second highest accuracy score for the data set as, C4.5 (90%), logistic regression (93%), Random forest (95.9%) and NB (96.3%), LVQ (97.3%), KNN (97.5%).