MEME17003 – APPLIED MULTIVARIATE ANALYSIS

Master of Mathematics

NOVEMBER 29, 2021

POOJA VIJAYAKUMAR
2106586

The problem I am interested in exploring is in developing K-Means clustering that can be used in predicting if the breast cancer tumor cell present in an individual is benign or malignant as well as using the algorithm in determining variables that can be used in predicting malignancy of cancerous cells. Breast cancer is a deadly killer and about 1 in 8 women in the United States will develop breast cancer in her lifetime. Hence, if cancerous cells are detected from early stages, it would be easier to treat and manage the disease before it costs lives. Therefore, it is important to detect these cancerous cells and classify them as benign or malignant. The particular dataset I would be using to conduct my experimentation would be the Breast Cancer Wisconsin (Diagnostic) Dataset obtained from UCI Machine Learning repository. This dataset consists of 32 columns of information describing characteristics of nuclei present in image of breast mass. 10 features are computed for the nuclei in which the mean, standard error (_se) and the worst of each feature were calculated for nuclei in each image and hence the total variables describing cell nuclei would be 30.

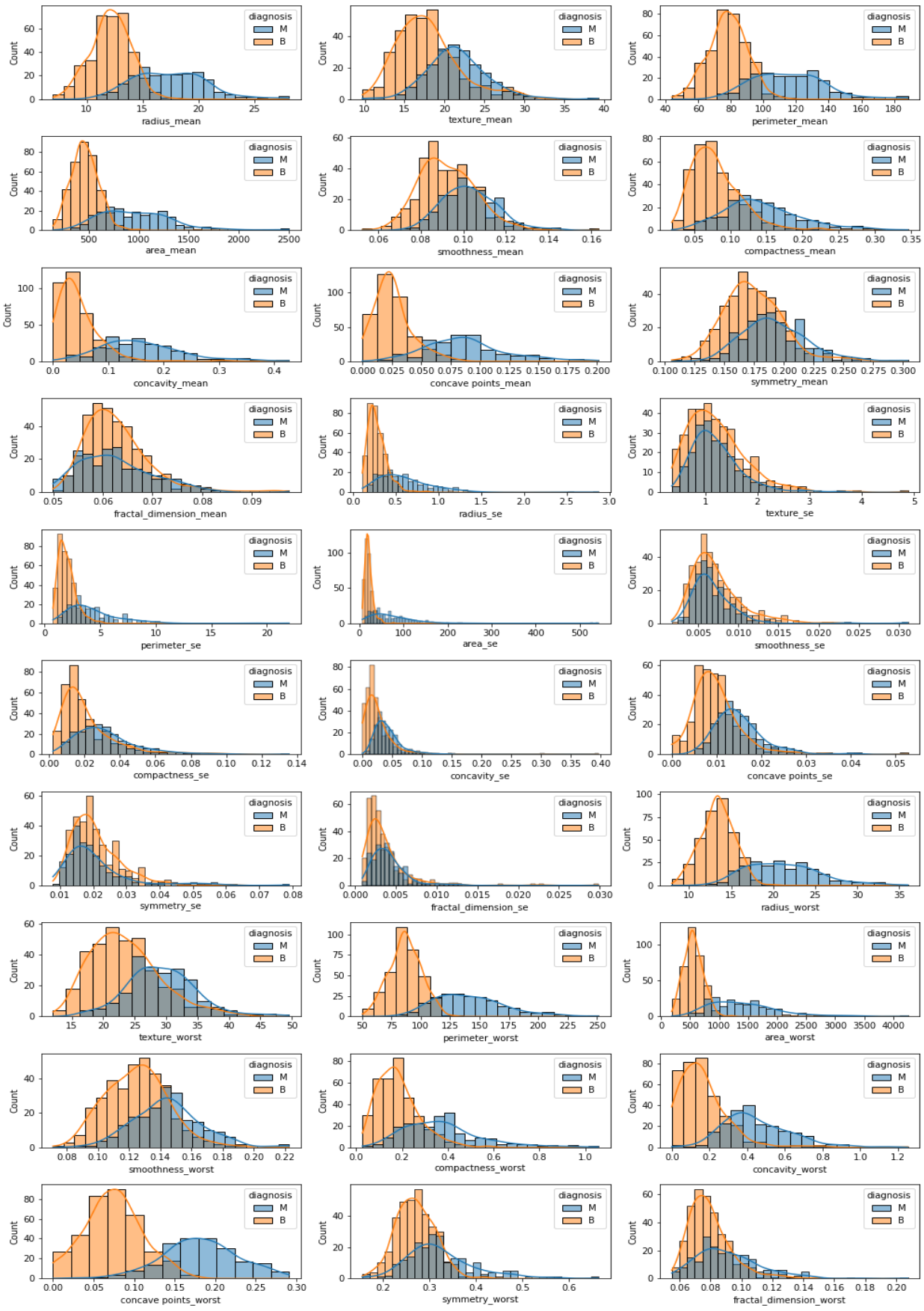
I would be using the K-Means clustering algorithm present in the Scikit-Learn library. The way the algorithms works by the following steps:

a. K centroids are randomly picked from sample data points and used as initial cluster centers.
b. Then, each data point is assigned to cluster whose centroid (mean) is nearest. The center of the cluster is then recalculated for each cluster.
c. Step b is repeated until no new assignments take place.

The distance metric used to calculate the similarity between each row of data and clusters is the Euclidean distance. One of the disadvantages of this algorithm usually lies in the fact that it is difficult to pre-determine the value of K before running the algorithm as we may not be sure how a set of data may be grouped. However, since in this case we would be classifying data based on if tumor cells are malignant or benign, it is easy to set the value of K to 2 as the data will be grouped into 2 clusters. The reason I selected the K-means clustering algorithm is that it is computationally simpler than hierarchical methods as the algorithm assigns data to a cluster based on just the Euclidean distance metric. It is also easy to implement the algorithm using the Scikit-Learn library.

First, each variable in the data is visualized to determine the distribution of the variables and see if there is separation of data into clusters from these variables. The variable ‘id’ is dropped as it does not provide any value in the clustering process as it is unique to each data point. The

variable ‘diagnosis’ is also dropped before passing the data to k-means algorithm as it contains the actual class (benign or malignant) of the tumor cell. However, this variable can later be used to determine the accuracy of the k-means algorithm so it is important to store it separately. Below shows the visualization of 30 variables.


diagnosis
M
B
Count
radius_mean
texture_mean
perimeter_mean
area_mean
smoothness_mean
compactness_mean
concavity_mean
concave points_mean
symmetry_mean
fractal_dimension_mean
radius_se
texture_se
perimeter_se
area_se
smoothness_se
compactness_se
concavity_se
concave points_se
symmetry_se
fractal_dimension_se
radius_worst
texture_worst
perimeter_worst
area_worst
smoothness_worst
compactness_worst
concavity_worst
concave points_worst
symmetry_worst
fractal_dimension_worst

Form the image above, it can be seen that there is clear distinction in data points between malignant and benign tumor for some of the variables. The distribution of data is more skewed to the left for diagnosis of benign and skewed to the right for diagnosis of malignant in many variables such as radius_mean, texture_mean and perimeter_mean. However, some variables show no separation of data for malignant and benign tumor such as variables texture_se and smoothness_se. This shows that proper feature selection may be useful in determining important variables to predict malignancy of cancerous cells. The k-means clustering algorithm is applied on the dataset to determine the predicted classes; benign or malignant for each of the data in the dataset. Then, the results obtained is then compared to the actual diagnosis and the accuracy is calculated.

The below shows the confusion matrix of the predictions compared to actual values with 0 indicating benign tumor and 1 indicating malignant tumor. It is shown that most of the classes were correctly classified into correct clusters with about 83 misclassifications.

```
confusion_matrix(y, y_pred)
```

```
array([[356,   1],
       [ 82, 130]], dtype=int64)
```

```
accuracy_score(y, y_pred)
```

```
0.8541300527240774
```

The accuracy of the predictions when calculated came to about 85%. This shows that the k-means algorithm is successful in clustering most of the data accurately into the correct tumor group. Hence, the clustering obtained appears to be valid. However, one of the disadvantages of the k-means clustering algorithm is that the results will often not be accurate if values of the variables are not standardized before values are passed into the algorithm. This is because when variables do not have similar units of measurement, the algorithm tends to put more importance towards some variables compared to others. Hence, the variables are standardized using min-max scaling which transforms the variables into a range of 0 to 1. Then, the algorithm is then applied again to the standardized data and the results are obtained.

The confusion matrix obtained when k-means algorithm is applied to the transformed data is as below. The confusion matrix shows that there is an increase in number of correctly classified rows of data. It appears more rows of data and assigned to the correct clusters.

```
confusion_matrix(y, y_pred)
```

```
array([[348,   9],
       [ 32, 180]], dtype=int64)
```

```
accuracy_score(y, y_pred)
```

```
0.9279437609841827
```

The accuracy has now jumped to about 93%. This shows that standardizing variables does have a positive impact on the performance of k-means algorithm on this dataset. By analyzing the distribution of variables in data as above, it can be observed that there are many variables that result in similar distribution for both benign and malignant cells; especially features with standard error. For example, the distribution of variable texture_se appears to be similar for both data with malignant and benign tumor cells. Moreover, the distribution of data in features with worst values appears to be similar to distribution of data in features with mean values.

Therefore, the variables with the standard errors and worst of the features were all removed. Hence, the variables left would be the mean values of each feature for each row of data. The k-means algorithm was then again fitted to the updated data and the accuracy were re-calculated.

```
confusion_matrix(y, y_pred)
```

```
array([[352,   5],
       [ 52, 160]], dtype=int64)
```

```
accuracy_score(y, y_pred)
```

```
0.8998242530755711
```

The algorithm managed to achieve an accuracy of about 90% with just the 10 variables; 'radius_mean', 'texture_mean', 'perimeter_mean', 'area_mean', 'smoothness_mean', 'compactness_mean', 'concavity_mean', 'concave points_mean', 'symmetry_mean' and 'fractal_dimension_mean'. This shows that the removed 20 variables does not provide significant information in segmenting the data into clusters as the accuracy score have just decreased by about 3% by removing them. Hence, when collecting new data to classify tumor cells, these variables could potentially be avoided.

Using the 10 remaining variables, the accuracy of the predictions of the algorithm will again be tested by removing 1 variable at a time. For each variable removed, the accuracy of the k-means algorithm by using the 9 remaining variables will be calculated. The results are displayed as below:

```
Removing column : radius_mean
Accuracy : 0.8910369068541301

Removing column : texture_mean
Accuracy : 0.9015817223198594

Removing column : perimeter_mean
Accuracy : 0.8927943760984183

Removing column : area_mean
Accuracy : 0.8998242530755711

Removing column : smoothness_mean
Accuracy : 0.9086115992970123

Removing column : compactness_mean
Accuracy : 0.9138840070298769

Removing column : concavity_mean
Accuracy : 0.9121265377855887

Removing column : concave points_mean
Accuracy : 0.9015817223198594

Removing column : symmetry_mean
Accuracy : 0.9068541300527241

Removing column : fractal_dimension_mean
Accuracy : 0.9033391915641477
```

The variables compactness_mean and concavity_mean result in the biggest increase in accuracy when removed. Hence the variables are removed from the dataset. The total accuracy when k-means algorithm was run on the remaining 8 variables; 'radius_mean', 'texture_mean', 'perimeter_mean', 'area_mean', 'smoothness_mean',, 'concave points_mean', 'symmetry_mean' and 'fractal_dimension_mean' for all the rows of data is about 91% as shown below.

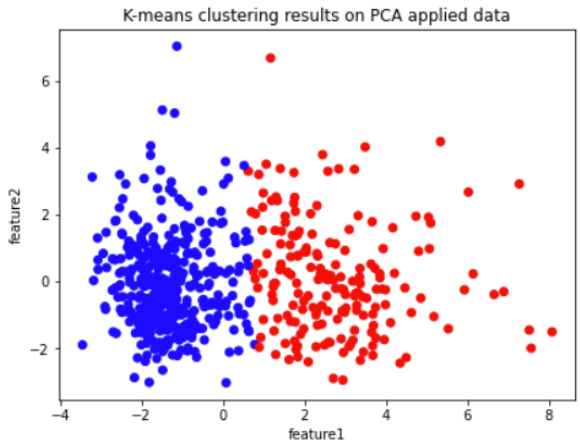
```
confusion_matrix(y, y_pred)
```

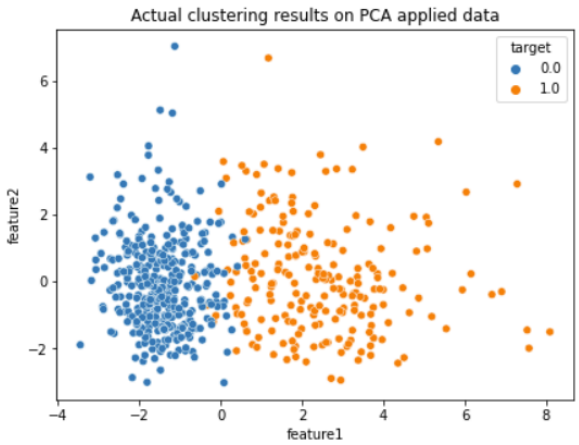
```
array([[356,   1],
       [ 51, 161]], dtype=int64)
```

```
print('Accuracy :',accuracy_score(y, y_pred))
```

```
Accuracy : 0.9086115992970123
```

Based on the accuracy and the results from confusion matrix, it is obvious that k-means does a good job in clustering the tumor cells into benign or malignant clusters with just 8 variables. However, without visualization, it is difficult to see the effectiveness of the developed k-means algorithm in clustering the data into the 2 classes. Hence, the dimension of the remaining 8 variables will be compressed and reduced to 2 using principal component analysis. This increases interpretability of the data and allows the data to be visualized in a 2-dimensional graph while minimizing information loss. Then, the result for k-means algorithm applied on the reduced data will be obtained and plotted together with the actual values as shown below:





As conclusion, a k-means algorithm that is able to cluster breast cancer tumor cell data into benign and malignant with high accuracy was developed. The best variables to correctly assign the data into correct clusters were identified to be 'radius_mean', 'texture_mean', 'perimeter_mean', 'area_mean', 'smoothness_mean',, 'concave points_mean', 'symmetry_mean' and 'fractal_dimension_mean'. Based on the results of visualization of data after applying principal component analysis algorithm and then applying k-means, it can be seen that there is a clear difference between the between distribution of data with malignant tumor cells and distribution of data with benign tumor cells. By comparing to the visualization of the actual clusters, it can be seen that k-means clustering algorithm is able to correctly cluster most of the data points accurately.

However, the weakness of the algorithm lies in the fact that it misclassifies data points that are close in proximity to the wrong cluster as belonging to that wrong cluster. Furthermore, while the accuracy of the results; 91% may seem high, it may be not be enough to be used in predicting life-or-death situation such as breast cancer. This is because there is still a 9% chance that tumor may be misclassified and may be fatal to patients. Hence, in this case, a 98% accuracy or more may be necessary. These issues can be mitigated by discovering new features from the cell nuclei to differentiate between benign and malignant tumor cells. We could also work on finding more data so that more accurate predictions can be made.