CW2: Data Mining the Diabetes Mellitus Database

Md Golam Rabby Shuvo, Reg. 100370344

May 18, 2022

Contents

1 Summary of Features **2**

2 Data Pre-processing **4**
2.1 Missing Data Handling 4
2.2 Categorical Label Encoding 4
2.3 Outlier Detection and Handling 5
2.4 Feature Selection 5

3 Supervised Model Training and Evaluation **6**
3.1 Sampling 7
3.2 Model Training 7
3.3 Evaluation 7

4 Unsupervised Clustering **10**
4.1 K-Means Clustering 10
4.2 Results 11

5 Conclusion **13**

A Appendix **14**

Abstract

The task of this coursework is basically data mining the diabetes mellitus dataset. Summary of the features available on the dataset is described at first. Then the cleaning of the dataset is done in the data pre-processing stage. After that, supervised model training and their evaluation is assessed. Lastly, unsupervised clustering is explored in the dataset. This report summarize all these contents. For supervised learning, Random Forest classifier yields higher accuracy of 87.21%. For unsupervised learning, 2 or 3 features could provide better clustering and good ARI and AMI scores.

1 Summary of Features

In this coursework, the given dataset is a part of a patient dataset with patients admitted to an ICU. The dataset has 79,160 observations and 88 columns. Among them the number of input features is 87. And, the target column is a particular type of diabetes, Diabetes Mellitus (diabetes_mellitus). In the given dataset, there are various information related to patient status in the ICU (demographics such as age, weight, BMI etc; APACHE-Acute Physiology and Chronic Health Evaluation covariates) and other related comorbidities; vital and laboratory test results collected within 24h of admission are provided. Here, the target column is a particular type of diabetes, Diabetes Mellitus.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 79159 entries, 0 to 79158
Data columns (total 88 columns):
 #   Column                 Non-Null Count  Dtype      #   Column                      Non-Null Count  Dtype
---  ------                 --------------  -----     ---  ------                      --------------  -----
 0   encounter_id           79159 non-null  int64      46  h1_hco3_max                 14808 non-null  float64
 1   hospital_id            79159 non-null  int64      47  h1_hco3_min                 14808 non-null  float64
 2   gender                 79129 non-null  object     48  h1_hematocrit_max           16616 non-null  float64
 3   ethnicity              78198 non-null  object     49  h1_hematocrit_min           16616 non-null  float64
 4   age                    76317 non-null  float64    50  h1_inr_max                  29693 non-null  float64
 5   elective_surgery       79159 non-null  int64      51  h1_inr_min                  29693 non-null  float64
 6   height                 77978 non-null  float64    52  h1_lactate_max              7174 non-null   float64
 7   weight                 77086 non-null  float64    53  h1_lactate_min              7174 non-null   float64
 8   bmi                    76468 non-null  float64    54  h1_sodium_max               17763 non-null  float64
 9   readmission_status     79159 non-null  int64      55  h1_sodium_min               17763 non-null  float64
 10  icu_type               79159 non-null  object     56  d1_arterial_po2_max         28333 non-null  float64
 11  h1_temp_max            61223 non-null  float64    57  d1_arterial_po2_min         28333 non-null  float64
 12  h1_temp_min            61223 non-null  float64    58  d1_pao2fio2ratio_max        22583 non-null  float64
 13  d1_albumin_max         36066 non-null  float64    59  d1_pao2fio2ratio_min        22583 non-null  float64
 14  d1_albumin_min         36066 non-null  float64    60  h1_arterial_pco2_max        13872 non-null  float64
 15  d1_bilirubin_max       32682 non-null  float64    61  h1_arterial_pco2_min        13872 non-null  float64
 16  d1_bilirubin_min       32682 non-null  float64    62  h1_arterial_ph_max          13770 non-null  float64
 17  d1_bun_max             71249 non-null  float64    63  h1_arterial_ph_min          13770 non-null  float64
 18  d1_bun_min             71249 non-null  float64    64  h1_arterial_po2_max         14017 non-null  float64
 19  d1_glucose_min         74800 non-null  float64    65  h1_arterial_po2_min         14017 non-null  float64
 20  d1_hco3_max            67258 non-null  float64    66  h1_pao2fio2ratio_max        10343 non-null  float64
 21  d1_hco3_min            67258 non-null  float64    67  h1_pao2fio2ratio_min        10343 non-null  float64
 22  d1_inr_max             29693 non-null  float64    68  wbc_apache                  61408 non-null  float64
 23  d1_inr_min             29693 non-null  float64    69  intubated_apache            79159 non-null  int64
 24  d1_lactate_max         21350 non-null  float64    70  d1_heartrate_max            79014 non-null  float64
 25  d1_lactate_min         21350 non-null  float64    71  heart_rate_apache           79003 non-null  float64
 26  d1_platelets_max       67993 non-null  float64    72  gcs_motor_apache            77935 non-null  float64
 27  d1_platelets_min       67993 non-null  float64    73  gcs_eyes_apache             77935 non-null  float64
 28  d1_potassium_max       71972 non-null  float64    74  creatinine_apache           64575 non-null  float64
 29  d1_potassium_min       71972 non-null  float64    75  bilirubin_apache            29109 non-null  float64
 30  d1_sodium_max          71544 non-null  float64    76  h1_spo2_max                 75355 non-null  float64
 31  d1_sodium_min          71544 non-null  float64    77  paco2_apache                18540 non-null  float64
 32  d1_wbc_max             68680 non-null  float64    78  map_apache                  78931 non-null  float64
 33  d1_wbc_min             68680 non-null  float64    79  aids                        79159 non-null  int64
 34  h1_albumin_max         6888 non-null   float64    80  cirrhosis                   79159 non-null  int64
 35  h1_albumin_min         6888 non-null   float64    81  hepatic_failure             79159 non-null  int64
 36  h1_bilirubin_max       6310 non-null   float64    82  immunosuppression           79159 non-null  int64
 37  h1_bilirubin_min       6310 non-null   float64    83  leukemia                    79159 non-null  int64
 38  h1_bun_max             15753 non-null  float64    84  lymphoma                    79159 non-null  int64
 39  h1_bun_min             15753 non-null  float64    85  solid_tumor_with_metastasis 79159 non-null  int64
 40  h1_calcium_max         15196 non-null  float64    86  ventilated_apache           79159 non-null  int64
 41  h1_calcium_min         15196 non-null  float64    87  diabetes_mellitus           79159 non-null  int64
 42  h1_creatinine_max      15884 non-null  float64   dtypes: float64(71), int64(14), object(3)
 43  h1_creatinine_min      15884 non-null  float64   memory usage: 53.1+ MB
 44  h1_glucose_max         35387 non-null  float64
 45  h1_glucose_min         35387 non-null  float64
```

Figure 1: Summary of the Features.

Figure 1 shows the summary information for the features which includes the number

of columns, column names, column data types, and the number of cells in each column (non-null values). From the 87 variables, there are 84 numerical variables and 3 categorical variables. The data types are *float64* (71 variables), *int64* (13 variables), *object* (3 variables). The missing data information can be found in Figure A.1 of Appendix A. From the percentage, 38 variables have more than 60% of missing data and rest of the 49 variables have less than 60% of missing data. By analysing correlation among the features, 2 pairs of variables have highly correlated data among them. One pair is (h1_inr_max, d1_inr_max) and other pair is (h1_inr_min, b1_inr_min). Both pairs have more than 60% of missing values. Figure 2 illustrates distribution of the diabates_mellitus column on whole dataset. Where 67% (≈50500) instances do not have diabates_mellitus, and rest of 33% (≈28700) instances have diabates_mellitus. The target variable does not have any missing data.

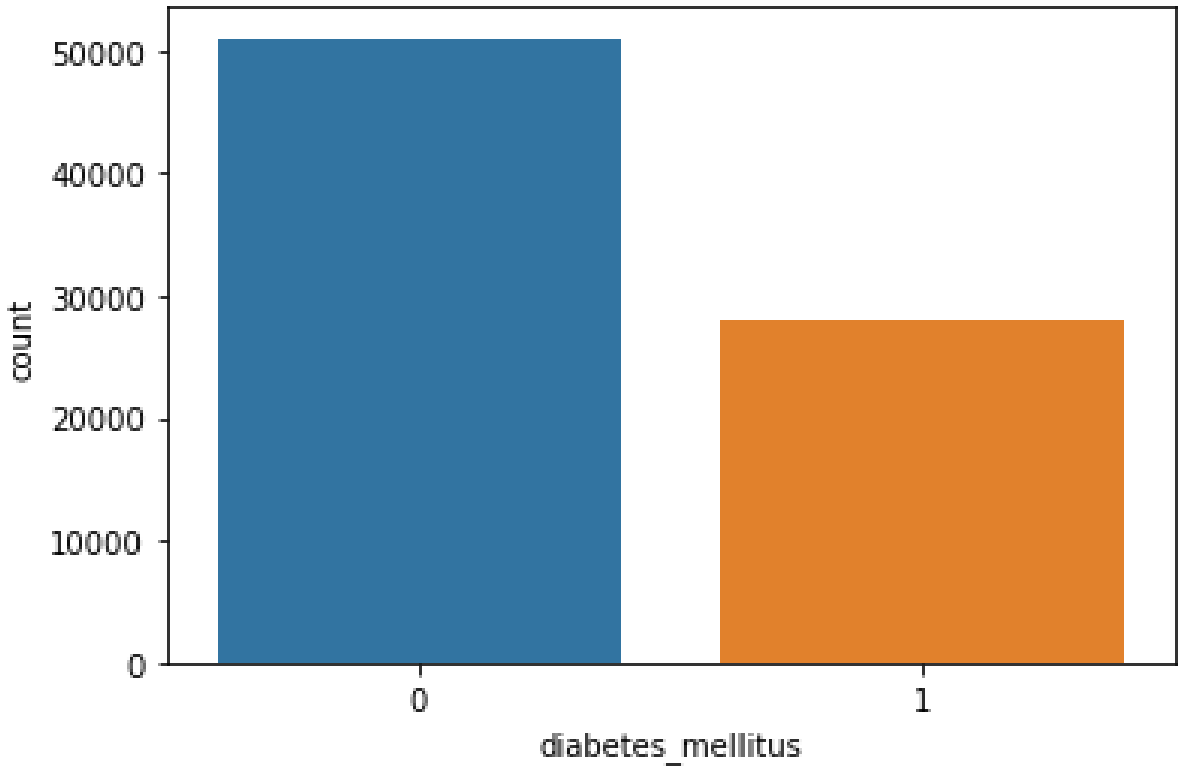

Figure 2: Distribution of target column over the dataset.

2 Data Pre-processing

In the data cleaning process many factors have been assessed, such as handling missing values, encoding of the categorical features, outlier detection and handling, and most important feature selection.

2.1 Missing Data Handling

From the missing percentage information in Figure A.1 of Appendix A, among all the 87 features there are 38 features comprises more than 60% of missing values. Thus, those features have been dropped from the dataset, and then 49 features are left. Though, there remains lots of missing data in other features too. Now, those missing values need to be imputed. As dataset consists categorical and numeric data. Missing data will be imputed based on their types. By creating a DataFrameIputer() class which takes the 'TransformerMixin' class from scikit-learn library and utilizing the class to impute missing data with the value after it and then with 0 for numeric columns and the most frequent value for non-numeric columns. After that, the new class to is used to transform the dataset and input the missing values. After handling missing values, there are no missing data left in the dataset. All the numerical missing values did not replace with 0 or mean values, as this imputation increases the mean of the each feature more than the current process.

2.2 Categorical Label Encoding

As our dataset has categorical features, those features need to be encoded to numerical values. Most of the algorithms in scikit-learn library works only for numerical data. The scikit-learn library provides category_encoders library to encode categorical vales. OrdinalEndcoder() has been used for the encoding process. Ordinal encoder changes each categorical label to a number, the columns are specified, and also it returns a dataframe. The values of three categorical features (gender, ethnicity, icu_type) encoded into numerical data, all of them have *float64* datatype. Therefore, our dataset is now consist of

only numerical data.

2.3 Outlier Detection and Handling

Many data mining projects begin with the discovery of outliers. The outlier identification process here aims to discover the parameters that are impacted by outlier tools from a large number of parameters. Detecting outliers may be accomplished using a variety of methods (Hsu et al., 2017). In the dataset, there could be so many outliers that can be harness accuracy result. So, outlier detection and handling is necessary for the better classification result.

IsolationForest algorithm is used for the outlier detection. It is a method for unsupervised learning that belongs to the family of ensemble decision trees. In lieu of profiling and creating normal points and areas, the method is designed to identify abnormalities. It exploits the fact that anomalies are the minority of data points and that their attribute-values are significantly different from those of regular cases. With the *contamination* parameter set to 0.01, it gives comparatively less outliers. Total number of outliers identified is 792. After that, outliers are handled by creating a mask to ensure only those rows that are not outliers are retained in a new data frame that can be used for classification. All the outliers have been removed from the dataset. The number of remaining observations is 78367.

2.4 Feature Selection

Now it is important task to determine which of the original features are better for the classification. For this, feature selection techniques need to be examined that independently rank features. Univariate feature selection is utilized here. Using univariate statistical tests, univariate feature selection selects the best features. It compares each attribute to the dependent variable to see whether or not they have a statistically significant connection (Pilyugina et al., 2021). It is also known as variance analysis (ANOVA). When analysing the link between one characteristic and the dependent variable, it disregard the other characteristics. This is why it is referred to as "univariate". Each element has its

own test score.

For ranking features f_classif function is used, which is the ANOVA F-value between labels for the classification tasks. F-scores and ranking of all features can be found in in Figure A.2 of Appendix A. Figure 3 displays the top 20 input variables based on their ranking, where all 20 features have F-score of more than the value of 60. Finally, all the features has been dropped from the dataset other than the top 20 features. Moreover, data pre-processing is done.

```
The top 20 features are:['ethnicity', 'age', 'weight', 'bmi', 'd1_bilirubin_max', 'd1_bilirubin_min', 'd1_bun_max', '
d1_bun_min', 'd1_glucose_min', 'd1_hco3_min', 'd1_platelets_min', 'd1_potassium_max', 'd1_potassium_min', 'd1_sodium_
min', 'h1_glucose_max', 'h1_glucose_min', 'd1_heartrate_max', 'gcs_motor_apache', 'gcs_eyes_apache', 'creatinine_apac
he']
```

Figure 3: Top 20 features based on the F-scores.

3 Supervised Model Training and Evaluation

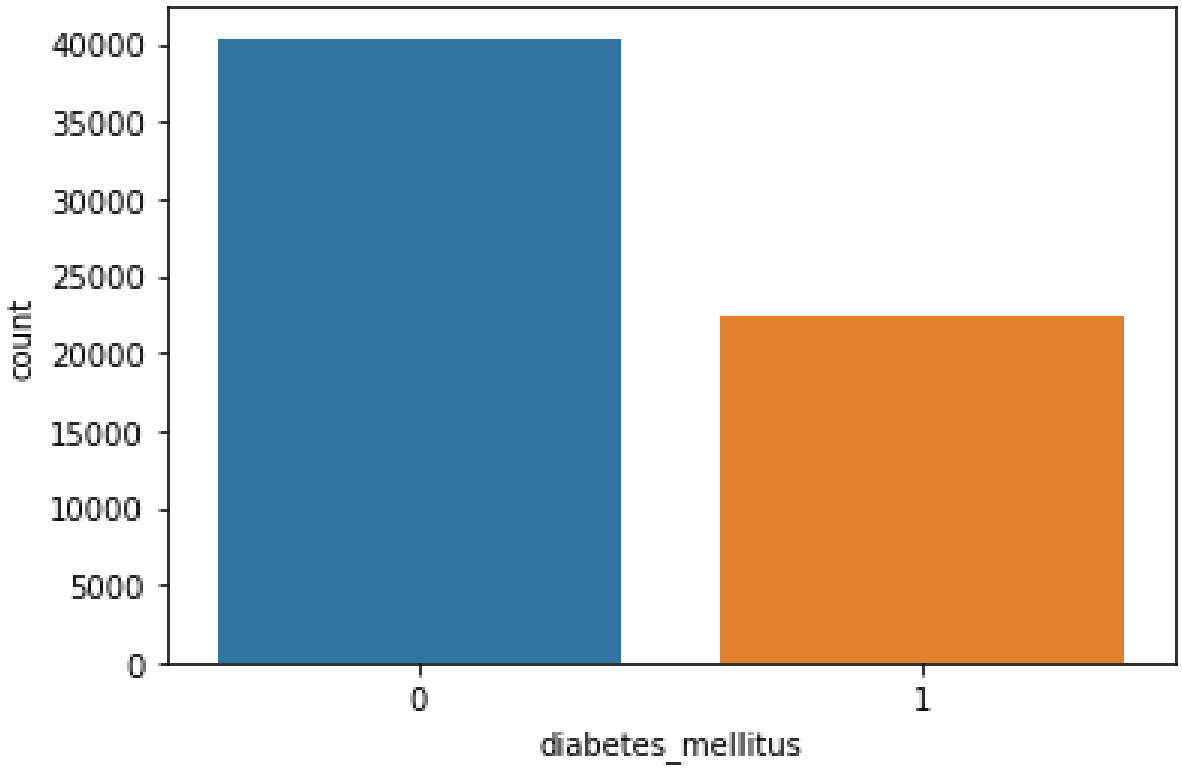

Figure 4: Distribution of target class over train dataset.

3.1 Sampling

For training the model, the first step is to split the dataset into train test data. Here, for this project, the dataset is split into ratio of 4:1 for the train and test data. Train set has 80% (62693) data and test set has 20% (15674). Now, the classes of the target column is not balanced. So, balancing the class is needed. Figure 4 shows the distribution of target class over training data. Majority of the observations do not have diabetes_mellitus. There are 40355 instances for without diabetes_mellitus and 22338 for with diabetes_mellitus. I have upsampled the minority class to balance the training data. After that, both classes have same instances (40355) over the train dataset.

3.2 Model Training

For the supervised model training, 3 simple classifiers that are commonly used in classification is trained. They are K-Nearest Neighbor (KNN), Support Vector Machine (SVM), and Random Forest classifier algorithms. As our target column is binary class, n_neighbors = 2 is selected for KNN. Kernel is the most important parameter in SVM classifier. Gaussian kernel is widely used because it depends on Euclidean Distance to measure similar points (Fischetti, 2016). For SVC, the kernel = gaussian is selected. And, for Random Forest classifier, I use n_estimators = 100. All the algorithm is utilized with the help of sklearn library. After building the training model for three classifiers, their results have been assessed further for finding the best classification algorithm.

3.3 Evaluation

Now comes the most important part, that is evaluation of the trained model. The evaluation is based on accuracy, f1 score, confusion matrix, and ROC curve of all the three classifiers.

Model accuracy is a most important performance indicator for supervised machine learning models that is stated as the proportion of true positives and true negatives to the total number of positive and negative events. On the other hand, using the harmonic mean, the F1-score combines the precision and recall of a classifier into a single statistic.

Classifier	Accuracy	F1 Score
K-Nearest Neighbor	0.7476	0.7317
Support Vector Machine	0.7001	0.6886
Random Forest	0.8721	0.8755

Table 1: Accuracy and F1 Score of three classifiers.

Primarily, it is used to test the efficiency of two classifiers. The Accuracy and F1 Score is better the higher it is, with 0 being the lowest possible result and 1 being the greatest. Table 1 shows the accuracy and f1 score for all three classifiers. Among them, Random Forest classifier gives higher accuracy of 87.21%. KNN and SVM give 74.76% and 70.01% of accuracy respectively. Random forest has also higher f1 score as well among them.

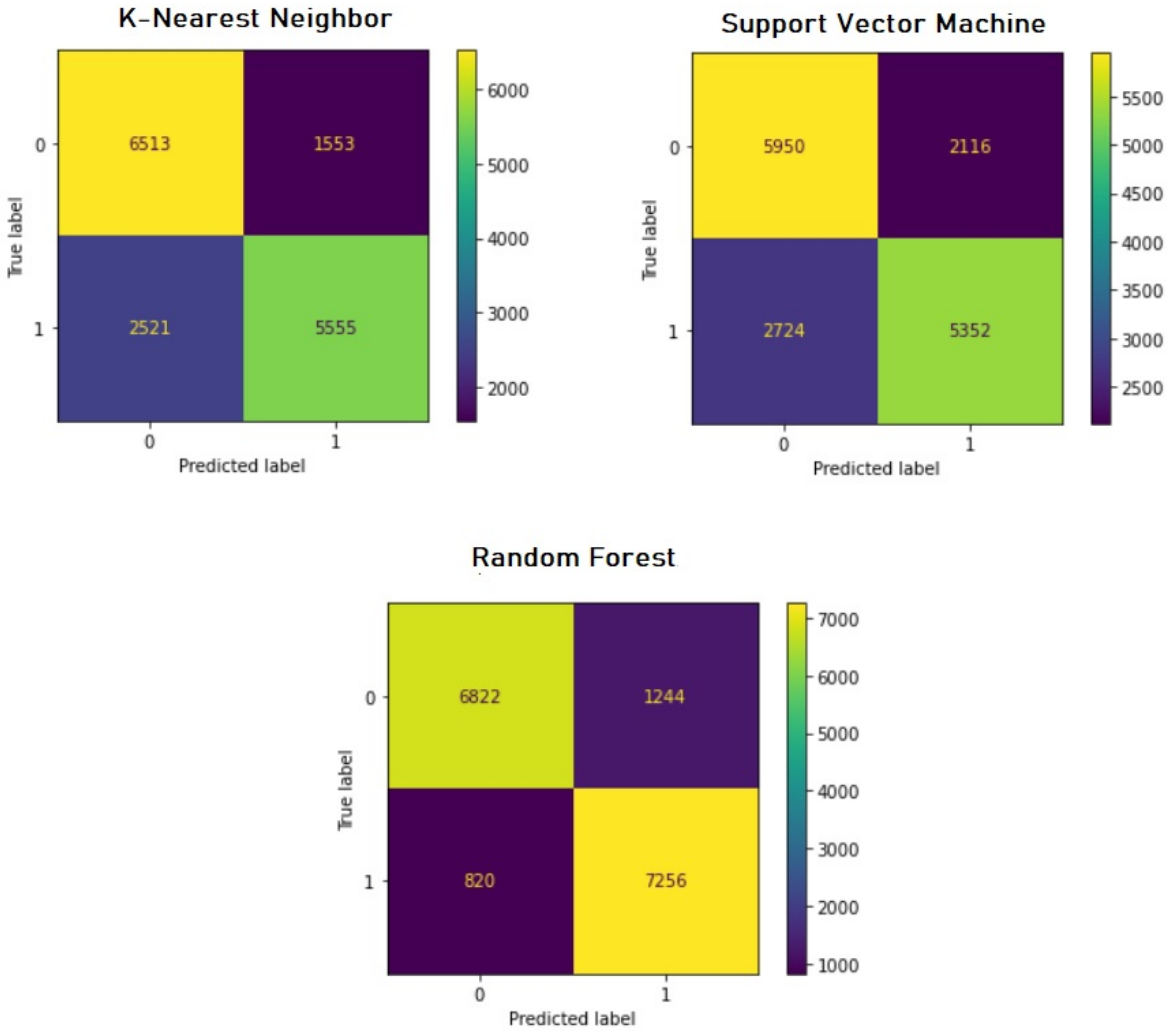

Figure 5: Confusion matrix for all three classifiers.

A confusion matrix is a summary of prediction results on a classification problem. The

number of correct and wrong predictions are summarized with count values and broken down by each class. The confusion matrix shows the ways in which your classification model is confused when it makes predictions (Luque et al., 2019). Figure 5 displays all the confusion matrix for the three classifier models. Random forest highlights better confusion matrix.

The precision-recall curve displays tradeoffs for various thresholds. A higher area under curve (AUC) implies equally high precision and high recall, while high precision means a low false positive rate and high recall means a low false negative rate. Figure 6 combines all the precision and recall curves for three classifiers. Random forest has perfect curve on the upper right corner with average precision (AP) = 0.96.

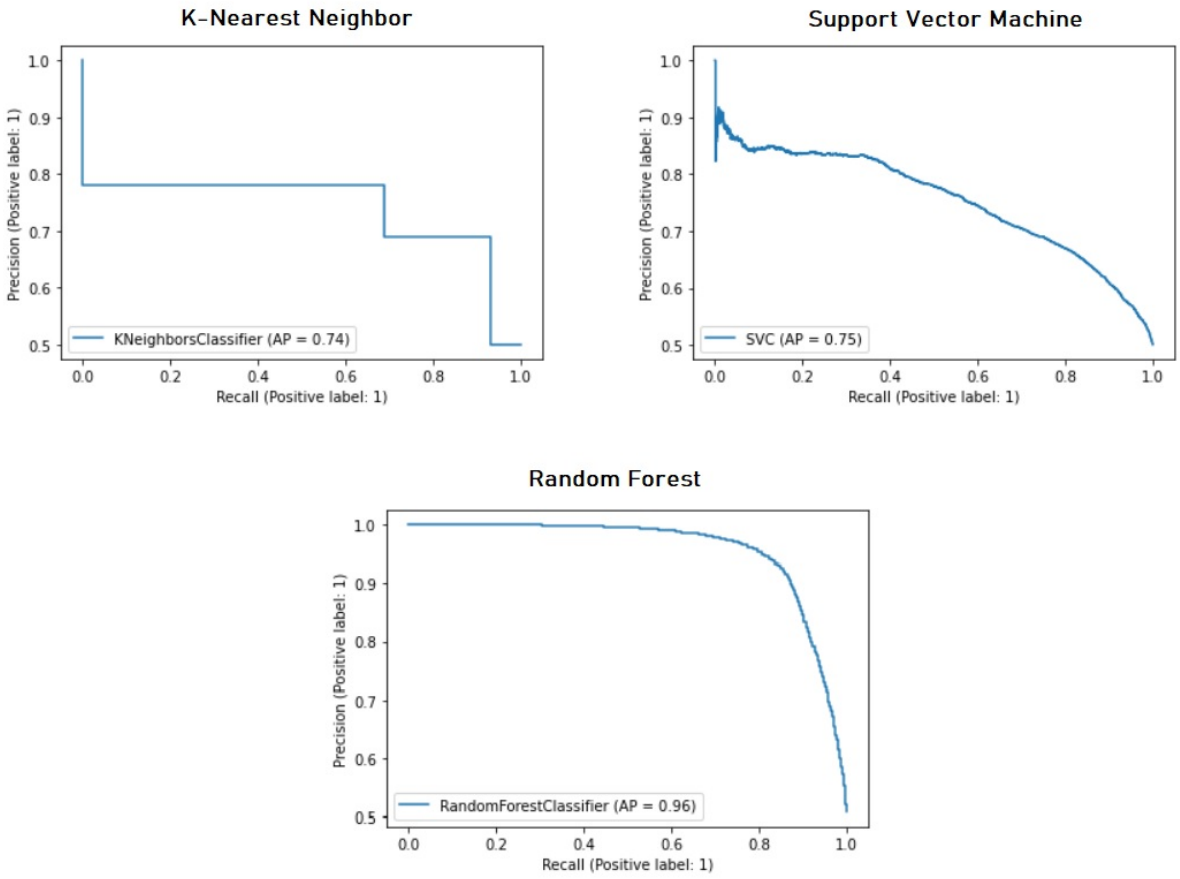

Figure 6: Precision-Recall curve for all three classifiers.

The Receiver Operating Characteristic (ROC) Curve is great way to visualize the performance of the multi-class classification. The ROC curve was created by plotting the true positive rate against the false positive rate at various threshold settings for the 5

targeted class. Figure 7 combines all the ROC curves for three classifiers. Random Forest reflects better ROC curve as its curve is the nearest to upper ledt hand corner. It has also higher area under curve (AUC) score of 0.95.

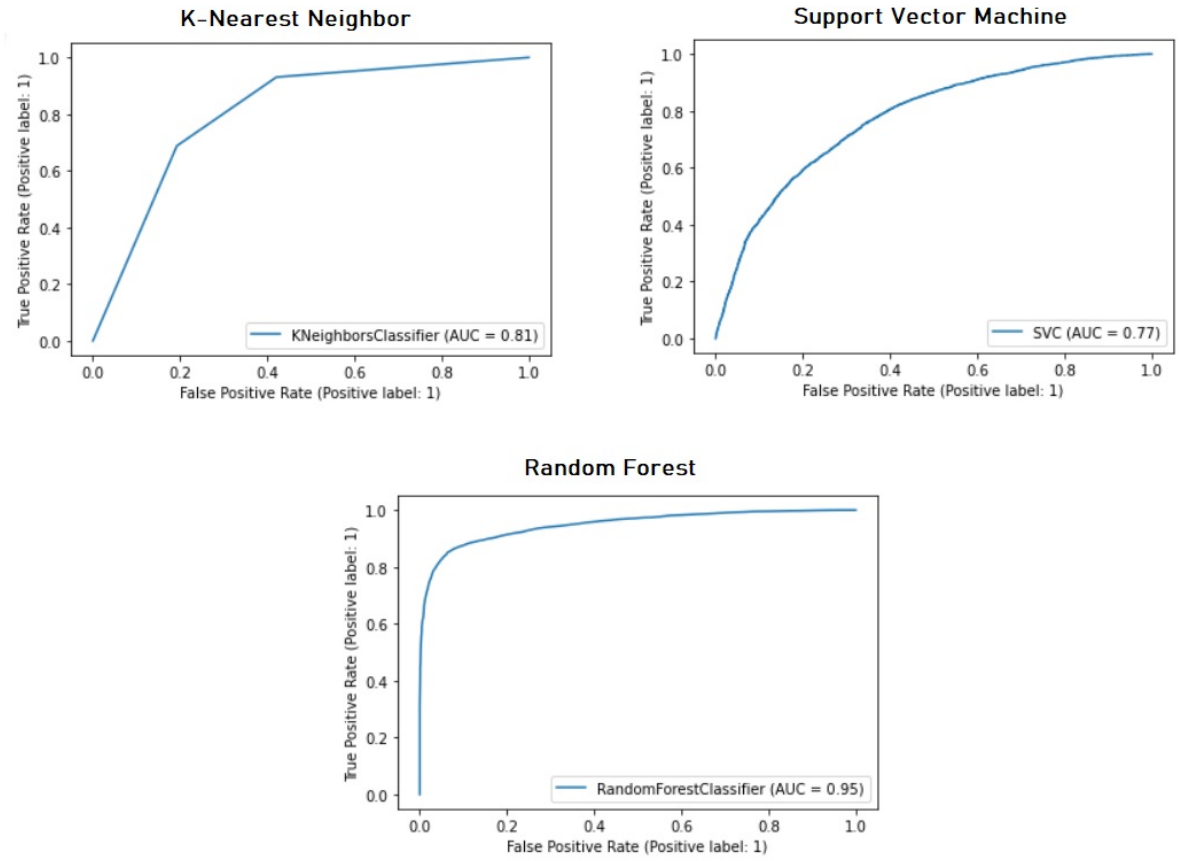

Figure 7: ROC curve for all three classifiers.

.

4 Unsupervised Clustering

4.1 K-Means Clustering

Unsupervised clustering algorithms are used against data which is not labelled. Thus, clustering can be done on only features dataset which excludes target column. For the unsupervised clustering, features dataset is reduced to half of the total instances. Now, the elbow method is utilized to find a good K-Means clustering for the features dataset. Initially clusters are considered in range of 1 to 10. Then, by plotting the silhouette score

of each of the kmeans score, the clusters are inspected better. Figure 8 combines the elbow plot and silhouette scores plot. Here the best clustering lies in number of clusters = 2.

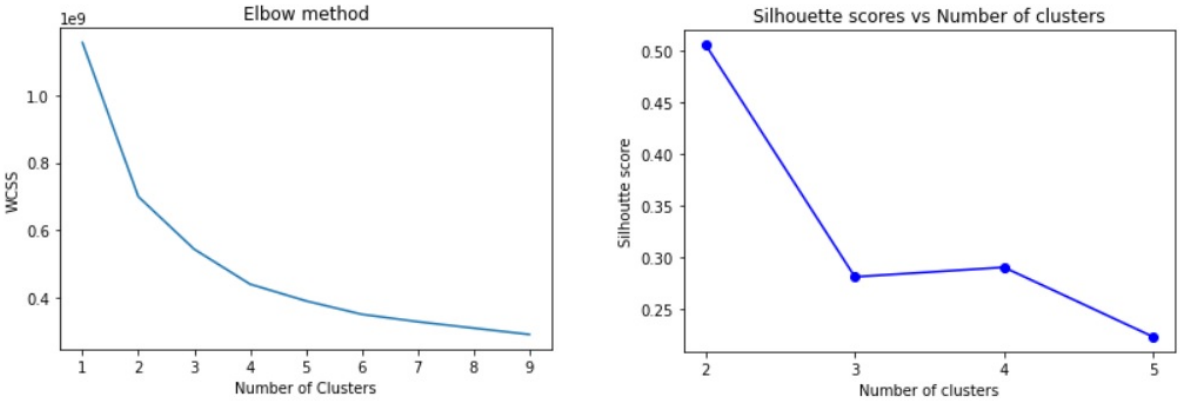

Figure 8: Elbow plot and Silhouette plot for clustering numbers.

Now, for visualization of clusters is not directly possible as our dataset has more than 3 features. However, by applying a Principal Component Analysis to reduce the space in 2 columns, it is possible to visualize the k-means clustering. Figure 9 shows the k-means clustering with 2 dimensions over half of our feature dataset.

4.2 Results

Fpr measuring the accuracy of Unsupervised learning methods, two scores have been used. One is Adjusted Rand Index (ARI) score, and another one is Adjusted Mutual Information (AMI) score. By comparing all sample pairings in the expected and true clusterings, the Rand Index computes a similarity measure between them. Using the following formula, the raw RI score is then 'adjusted for chance' to get the ARI score:

$$ARI = \frac{RI - Expected\_RI}{max(RI) - Expected\_RI}$$

AMI is a chance-adjusted Mutual Information (MI) score. It explains why the MI is greater for two clusterings with more clusters, regardless of whether more information is

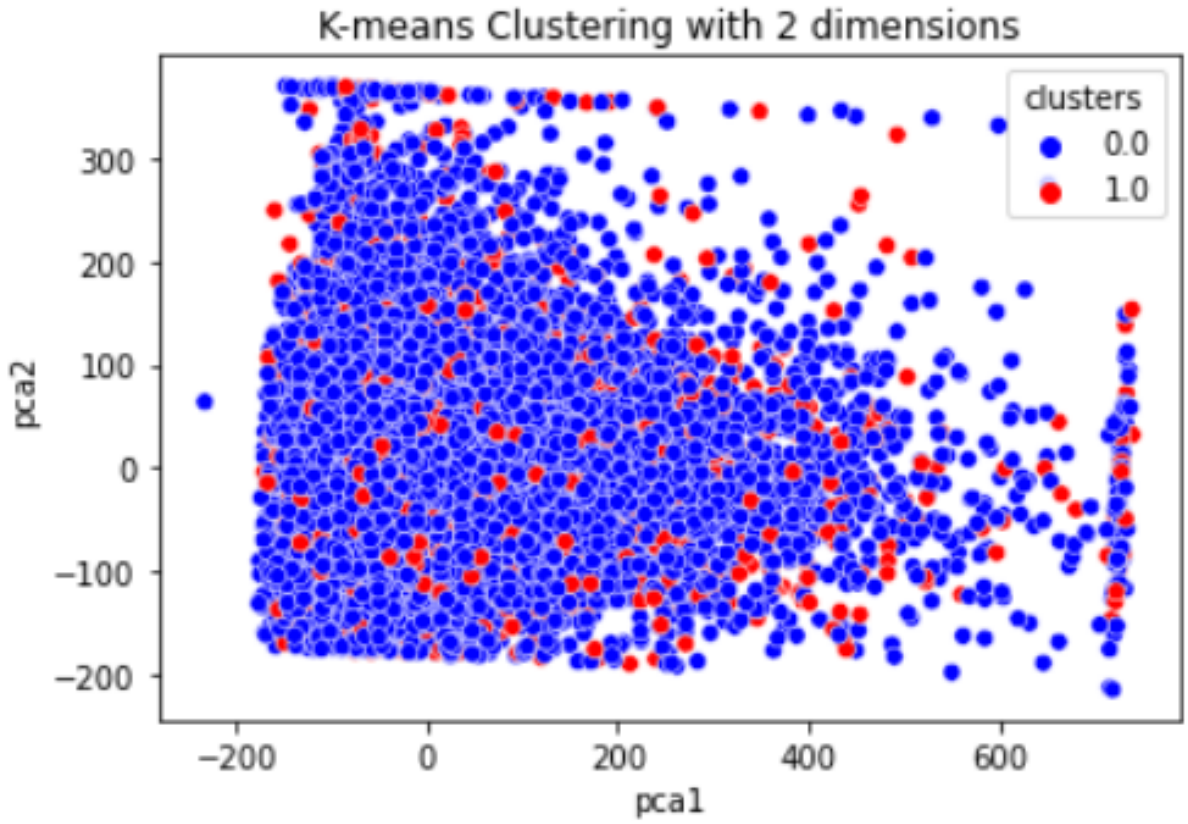

Figure 9: K-means clustering with 2 dimensions using PCA.

communicated. The AMI is calculated as follows for the two clusterings:

$$AMI(U,V) = \frac{MI(U,V) - E(MI(U,V))}{avg(H(U),H(V)) - E(MI(U,V))}$$

Table 2 stores the ARI and AMI score of the K-means clustering. Both scores are pretty low. As, clustering is done on 20 features, clustering does not reflect simple clusters. That's why scores are also bad for this unsupervised clustering.

Algorithm	ARI Score	AMI Score
K-means Clustering	0.118736	0.074001

Table 2: ARI and AMI score of the k-means clustering.

5 Conclusion

The main objective of this coursework lies in the classification of the diabetes mellitus. The dataset had 87 features with numerical and categorical variables at first. For handling the missing data 38 features is dropped in the cleaning process. Among 39 features, top 20 features is selected based on univariate feature selection. In the data pre processing stage, categorical labels are encoded to numerical values and outliers are handled for better results.

From the remained observations of the dataset and using three classifier algorithms, the prediction for the dataset is made in supervised model training; and evaluation is reported with the help of accuracy, f1 score, confusion matrix, precision-recall curve, and ROC curve. Random Forest classifier algorithm gives the highest accuracy of 87.21% among them. Moreover, from all the classification performance metrics evaluation, Random Forest classifier perform better for our dataset. In the last stage, unsupervised clustering is improvised. The number of clusters in K-means clustering is achieved with the help of elbow method and silhouette scores. For visualization of K-means clustering PCA has been used. However, the features dataset has so many features, the clusters are not grouped simply. Thus, ARI and AMI score for the K-means clustering are so low here. However, K-means clustering with 2 or 3 features could give better ARI and AMI scores.

References

Fischetti, M. (2016). Fast training of support vector machines with gaussian kernel. *Discrete Optimization*, 22:183–194. SI: ISCO 2014.

Hsu, H.-H., Chang, C.-Y., and Hsu, C.-H. (2017). Chapter 13 - the internet of things and its applications. In *Big Data Analytics for Sensor-Network Collected Intelligence*, Intelligent Data-Centric Systems, pages 256–279. Academic Press.

Luque, A., Carrasco, A., Martín, A., and de las Heras, A. (2019). The impact of class imbalance in classification performance metrics based on the binary confusion matrix. *Pattern Recognition*, 91:216–231.

Pilyugina, N., Tsukahara, A., and Tanaka, K. (2021). Comparing methods of feature extraction of brain activities for octave illusion classification using machine learning. *Sensors (Basel, Switzerland)*, 21(19):6407.

A Appendix

	Percent_missing
h1_bilirubin_min	92.03
h1_bilirubin_max	92.03
h1_albumin_min	91.30
h1_albumin_max	91.30
h1_lactate_min	90.94
h1_lactate_max	90.94
h1_pao2fio2ratio_max	86.93
h1_pao2fio2ratio_min	86.93
h1_arterial_ph_max	82.60
h1_arterial_ph_min	82.60
h1_arterial_pco2_min	82.48
h1_arterial_pco2_max	82.48
h1_arterial_po2_min	82.29
h1_arterial_po2_max	82.29
h1_hco3_max	81.29
h1_hco3_min	81.29
h1_calcium_min	80.80
h1_calcium_max	80.80
h1_bun_min	80.10
h1_bun_max	80.10
h1_creatinine_max	79.93
h1_creatinine_min	79.93
h1_hematocrit_min	79.01
h1_hematocrit_max	79.01
h1_sodium_min	77.56
h1_sodium_max	77.56
paco2_apache	76.58
d1_lactate_min	73.03
d1_lactate_max	73.03
d1_pao2fio2ratio_max	71.47
d1_pao2fio2ratio_min	71.47
d1_arterial_po2_min	64.21
d1_arterial_po2_max	64.21
bilirubin_apache	63.23
d1_inr_min	62.49
h1_inr_max	62.49
d1_inr_max	62.49
h1_inr_min	62.49
d1_bilirubin_max	58.71
d1_bilirubin_min	58.71
h1_glucose_min	55.30
h1_glucose_max	55.30
d1_albumin_max	54.44
d1_albumin_min	54.44

	Percent_missing
h1_temp_max	22.66
h1_temp_min	22.66
wbc_apache	22.42
creatinine_apache	18.42
d1_hco3_min	15.03
d1_hco3_max	15.03
d1_platelets_min	14.11
d1_platelets_max	14.11
d1_wbc_max	13.24
d1_wbc_min	13.24
d1_bun_min	9.99
d1_bun_max	9.99
d1_sodium_min	9.62
d1_sodium_max	9.62
d1_potassium_min	9.08
d1_potassium_max	9.08
d1_glucose_min	5.51
h1_spo2_max	4.81
age	3.59
bmi	3.40
weight	2.62
gcs_motor_apache	1.55
gcs_eyes_apache	1.55
height	1.49
ethnicity	1.21
map_apache	0.29
heart_rate_apache	0.20
d1_heartrate_max	0.18
gender	0.04
cirrhosis	0.00
lymphoma	0.00
immunosuppression	0.00
ventilated_apache	0.00
solid_tumor_with_metastasis	0.00
leukemia	0.00
hepatic_failure	0.00
encounter_id	0.00
aids	0.00
intubated_apache	0.00
hospital_id	0.00
icu_type	0.00
readmission_status	0.00
elective_surgery	0.00
diabetes_mellitus	0.00

Figure A.1: Missing data percentage of all features.

	Column_names	F_Scores
30	h1_glucose_max	9206.020775
31	h1_glucose_min	8547.156670
8	bmi	2950.884105
7	weight	2562.326530
17	d1_bun_max	2074.913764
18	d1_bun_min	1796.154716
19	d1_glucose_min	1708.625941
38	creatinine_apache	1527.258684
24	d1_potassium_max	901.790606
4	age	731.678701
27	d1_sodium_min	278.024870
15	d1_bilirubin_max	225.806802
16	d1_bilirubin_min	202.295362
21	d1_hco3_min	192.618066
3	ethnicity	181.492371
25	d1_potassium_min	128.669742
34	d1_heartrate_max	84.228514
37	gcs_eyes_apache	68.349387
36	gcs_motor_apache	67.764117
23	d1_platelets_min	63.077647
22	d1_platelets_max	52.902362
35	heart_rate_apache	44.064173
26	d1_sodium_max	41.038477
14	d1_albumin_min	37.014174
47	solid_tumor_with_metastasis	33.680268
13	d1_albumin_max	29.664902
42	cirrhosis	22.639192
39	h1_spo2_max	19.621915
48	ventilated_apache	11.819840
43	hepatic_failure	11.453264
12	h1_temp_min	8.420247
41	aids	7.580258
2	gender	6.019383
29	d1_wbc_min	5.537472
11	h1_temp_max	5.383852
20	d1_hco3_max	4.844056
32	wbc_apache	3.249259
5	elective_surgery	2.277927
10	icu_type	2.182276
44	immunosuppression	1.613670
6	height	0.836971
45	leukemia	0.617601
28	d1_wbc_max	0.564669
46	lymphoma	0.468355
40	map_apache	0.267360
1	hospital_id	0.183523
33	intubated_apache	0.088500
0	encounter_id	0.011329
9	readmission_status	NaN

Out[22]: (78367, 49)

Figure A.2: F-scores and ranking of all features.