Diabetes Insights & Trends (Saudi vs International Data)

February, 2021

DSIVIII

Team Green Elbow:

Abdulaziz Alsulami
Amjaad Alsubaie
Najwa Alsaeede
Ahmad Adam

CONTENTS

ABSTRACT

Background: In Saudi Arabia, diabetes is one of the most prevalent diseases impacting the quality of life of many individuals and causing an immeasurable health and financial burden on the country's economy.

Objectives: To explore the prevalence, common comorbidities and predictive factors of diabetes among the Saudi population and build a machine learning model that identifies undiagnosed diabetic individuals.

Methods: The data was obtained from Lean Business Services and contains medical records of diabetes within Saudi Arabia and internationally. The analytical approach regarding the identification of diabetic patients is binary in nature and hence binary classification techniques and models were employed.

Results: The best diabetes predicting models were achieved using the Cat Boosting Classifier and the Gradient Boosting Classifier in the local and international data (F1-scores: 0.747 & 0.715) with class rebalance techniques respectively.

Conclusion: The numerous illogical (mismatch) values in the data may have negatively contributed to model accuracy. Due to the sensitivity of medical data, filling/replacing such values would come at the expense of data integrity. It is therefore recommended to consult bioinformatician regarding best practice at handling medical data.

Introduction

Diabetes mellitus, commonly known as diabetes, is metabolic disorder characterized by the presence of high blood sugar levels over a long period of time[1]. Symptoms of diabetes include frequent urination, increased thirst and increased hunger levels. Untreated diabetes can result in serious long-term complications including cardiovascular disease, stroke, kidney disease, nerve and eye damage, cognitive impairment and death[1].

Type 1 and Type 2 are the most common types of diabetes. Type 1 diabetes results from the failure of the pancreas to produce enough insulin due to loss of insulin-producing cells. This form is usually present from birth and is believed to be caused by various genetic and autoimmune factors. On the other hand, Type 2 diabetes, commonly recognized as adult-onset diabetes, is characterized by an inadequate insulin response by cells. The causes of Type 2 diabetes are largely lifestyle-dependent including excessive body weight and insufficient exercise[2].

In 2019 alone, 1.5 million people have died due to diabetes and estimates indicate that 463 million people are living with diabetes all over the world[3]. According to The World Health Organization, Saudi Arabia has the second highest rate of diabetes in the Middle East, and is seventh in the world for the rate of diabetes. Recent estimates indicate around 7 million of the Saudi population are diabetic and around 3 million have pre-diabetes[4].

In Saudi Arabia, the associated health and economic burden due to diabetes is predicted to rise significantly emphasizing the need for urgent control measures such as rising public awareness towards the importance of adopting a healthy and active lifestyle. However, for optimum implementation of such control measures, effective healthcare documentation systems must be utilized to gather accurate data and facilitate action.

Here, we explore the prevalence of diabetes, examine the comorbidities and predictive factors of diabetes among the Saudi population. Our goal is to improve the quality of life by utilizing the power of artificial intelligence. Our objective is to build a machine learning model that identifies undiagnosed diabetic individuals in Saudi Arabia.

Data joining

Local data

This project draws insights and conclusions from two data origins, local and international, each of which has several data files (**Tables 1 & 3**).

The local data constitutes 4 files with various shapes (**Table 1**). The Services file was very large (~140 million records) and was hence segmented into 14 chunks of 10 million records each. These were joined together into one dataset using a common column (Patient ID) **Table 2**.

Encounters and Diagnosis data files were joined on the common column (U_Encouter_ID) resulting in the file (Merge 1). Merge 1 was subsequently joined with the Sehaty data file on the common column (PatientID) resulting in the file (Merge 2). Finally, Merge 2 was joined with the Service's (Chunk 1) file resulting in the final local dataset of 298,514 records. Merging details of all local data files are outlined in **Table 2.**

Table 1. Summary of local data files

File Name	No Rows	No Columns
Sehaty	982,044	12
Diagnosis	8,279,099	5
Encounters	20,823,172	30
Services (Chunk1)	10,000,000	13

Table 2. Local data files merge summary

Merge No	File1	File2	Merged on	No Rows	No Columns
1	Encounters	Diagnosis	U_Encouter_ID	8,279,099	16
2	Merge-1	Sehaty	PatientId	298,514	24
3	Merge-2	Service's (Chunk 1)	PatientId	298,514	28

International data

All international data files were merged on the common column (patientId) resulting in a final data shape of 19939 records and 30 columns (**Table 4**). Generally, the international data was mainly utilized for comparison purposes of useful diabetes predictors and disease comorbidities.

Table 3. Summary of international data files

File Name	No Rows	No columns
Diagnosis_data	9948	6
patient_data	9948	9
predicting_results	19939	14

Table 4. International data files merge summary

Merge No	File1	File2	Merged on	No Rows	No Columns
1	Diagnosis _data	patient_data	PatientId	9948	14
2	Merge-1	predicting_results	PatientId	19939	30

Methodology

Business Understanding

Problem statement

The importance of accurate healthcare documentation is vital for any functional healthcare system. It is through effective health documentation systems that diseases can be predicted and managed with minimum health and economic costs.

In Saudi Arabia, diabetes is one of the most prevalent diseases impacting the quality of life of many individuals and causing an immeasurable financial burden on the country's economy.

Objectives

- To explore the prevalence of diabetes in Saudi Arabia.
- To explore common comorbidities and predictive factors of diabetes among the Saudi population
- To identify healthcare utilization among diabetic patients
- To build a machine learning model that identifies undiagnosed diabetic individuals.

Analytic Approach

Provided data includes a list of patients with various features, some of which can be used to identify common comorbidities, prevalence/incidence of diabetes in Saudi Arabia.

The most important feature (whether a patient has diabetes or not) can be utilized as a target for binary classification model. According to given features, the model will predict the probability of diabetes of undiagnosed individuals

Data Source and requirements

The data was provided by Lean Business Services, a well-known health service provider with immense impact on health data within the kingdom.

Hence, the content, formats, and data representations were therefore prepared in accordance with Lean requirements. Domain knowledge experts were consulted as needed.

Data understanding

Descriptive statistics and visualization techniques were used to assess the quality of the data and gain initial insights into the data.

Local data

After merging the data into one file (shape: 298,514 / 28), there were many duplicate values on the patient ID column (multiple visits per patient). Since diabetes is our target, it is important not to delete any patient ID where the target is true.

Hence, a for loop was generated to rule out any ambiguity in duplicated patient IDs. In the below code, we double checked that no duplicated patient ID meets the condition where in one visit (the target is true) and in another visit (the target is false). This insured that removing duplicated patient ID did not affect the integrity of our target (Patient is diabetic).

As such, we adopted the most updated visit per patient (last patient visit) and removed the remaining visits. This is appropriate because the last visit constitutes the most updated patient status. The resulting shape after removing duplicated patient visits was (94516 / 18).

```
In [50]: has_diabetes = []
         has_no_diabetes = []

         for i in (df[df['Patient_ID'].duplicated()].index):
             if df.loc[i,'has_diabetes'] == True:
                 has_diabetes.append(i)
             elif df.loc[i, 'has_diabetes'] == False:
                 has_no_diabetes.append(i)

In [51]: has_diabetes_and_has_no_diabetes = []
         for n in lst1:
             if n in lst2:
                 has_diabetes_and_has_no_diabetes.append(n)

In [52]: has_diabetes_none_diabetes
Out[52]: []
```

International data

All international data files were merged on the common column (patientId) resulting in a final data shape of 19939 records and 30 columns. Generally, the international data was mainly utilized for comparison purposes of useful diabetes predictors and disease comorbidities.

Visualization

Local data

```
In [23]: 1 df_clean1.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 94602 entries, 0 to 94601
Data columns (total 28 columns):
 #   Column                 Non-Null Count  Dtype
---  ------                 --------------  -----
 0   Patient_ID             94602 non-null  int64
 1   U_Encounter_ID_x       94602 non-null  object
 2   Service Type           93157 non-null  object
 3   Service_Code           83632 non-null  object
 4   Service Name           90520 non-null  object
 5   Quantity               94602 non-null  int64
 6   Service Date           87396 non-null  datetime64[ns]
 7   VitalSigns             87937 non-null  object
 8   U_Encounter_ID_y       94602 non-null  object
 9   Provider Name          94602 non-null  object
 10  Admission Date/Time    94596 non-null  object
 11  Discharge Date/Time    85593 non-null  object
 12  Encounter_Type         94602 non-null  object
 13  Duration of leave      85587 non-null  float64
 14  Date of Birth          94493 non-null  object
 15  Gender                 94602 non-null  object
 16  Nationality            94515 non-null  object
 17  Discharge Mode         59611 non-null  object
 18  Clinic                 94600 non-null  object
 19  MaritalStatus          51093 non-null  object
 20  Diagnosis Description  94602 non-null  object
 21  Diagnosis Code         94602 non-null  object
 22  Diagnosis Type         94602 non-null  object
 23  height                 46634 non-null  float64
 24  weight                 47216 non-null  float64
 25  gender                 94602 non-null  object
 26  has_diabetes           94602 non-null  bool
 27  has_hypertension       87808 non-null  object
dtypes: bool(1), datetime64[ns](1), float64(3), int64(2), object(21)
memory usage: 19.6+ MB
```

Fig. General information of the local data

Visualization of important columns

1. (has_diabetes)

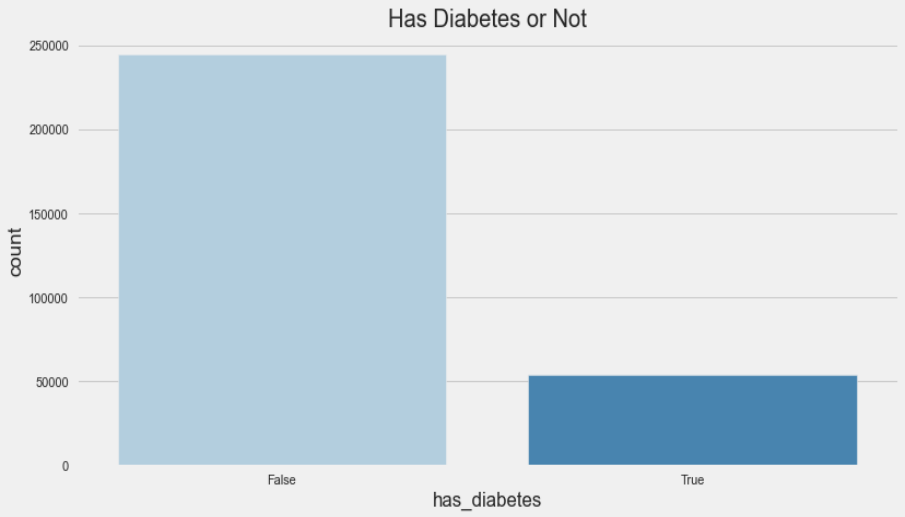


Fig. Illustration of whether patient has diabetes or not. Most patients do not suffer from diabetes.

2. Gender

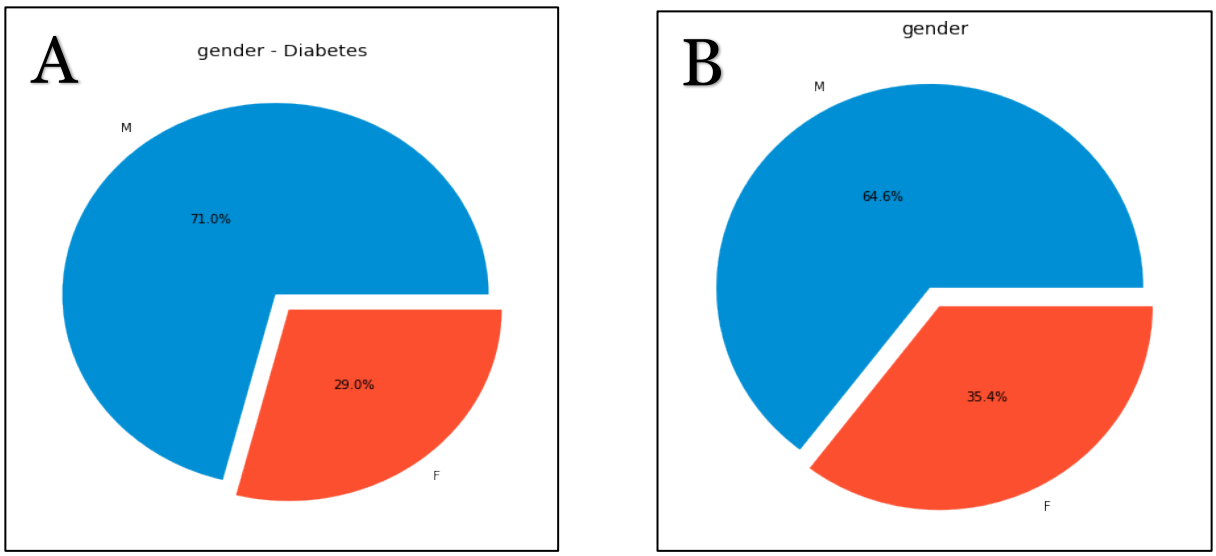


Fig. Pie chart representation of gender for all and diabetic patients. Diabetes incidents are relatively more frequent in males compared to females (A). The data contains more males compared to females (B).

3. Diagnosis Description

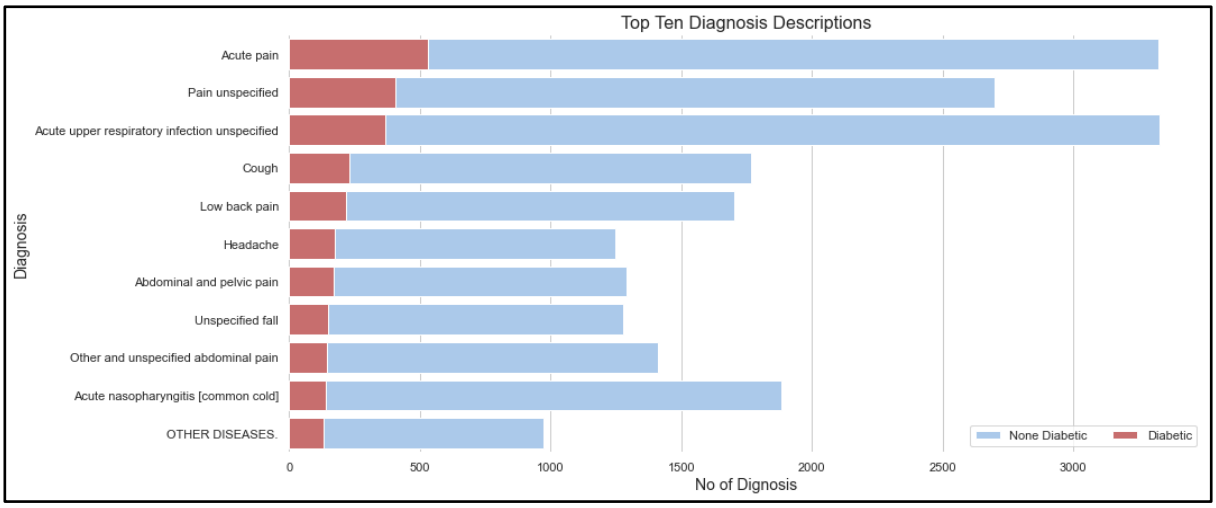


Fig. Top ten diagnosis descriptions - diabetic vs none-diabetic. Pain is the most form of complain by both diabetic and none-diabetic patients.

4. has_hypertension

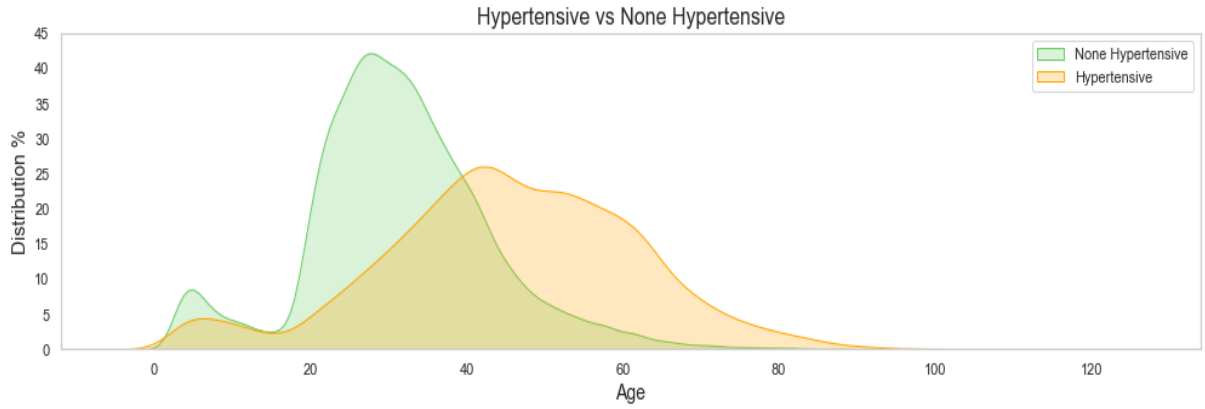


Fig. Age distribution of hypertensive and none-hypertensive patients. There is a correlation of hypertension with age. Those aged 40 - 60 are high risk group for developing hypertension.

5. Nationality

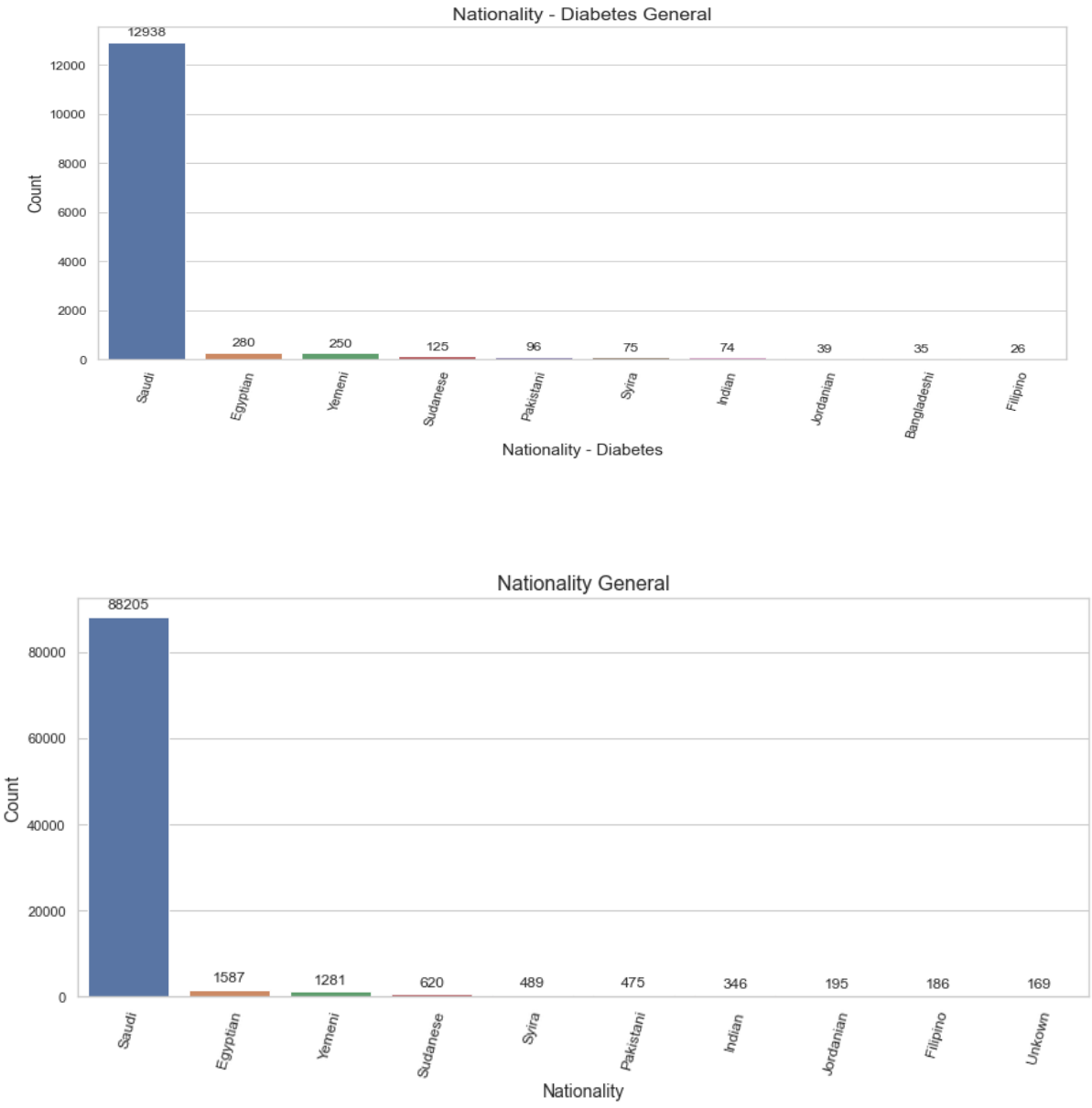


Fig. Nationality demographics. Saudi citizens represent over 90% of all nationalities.

6. Age

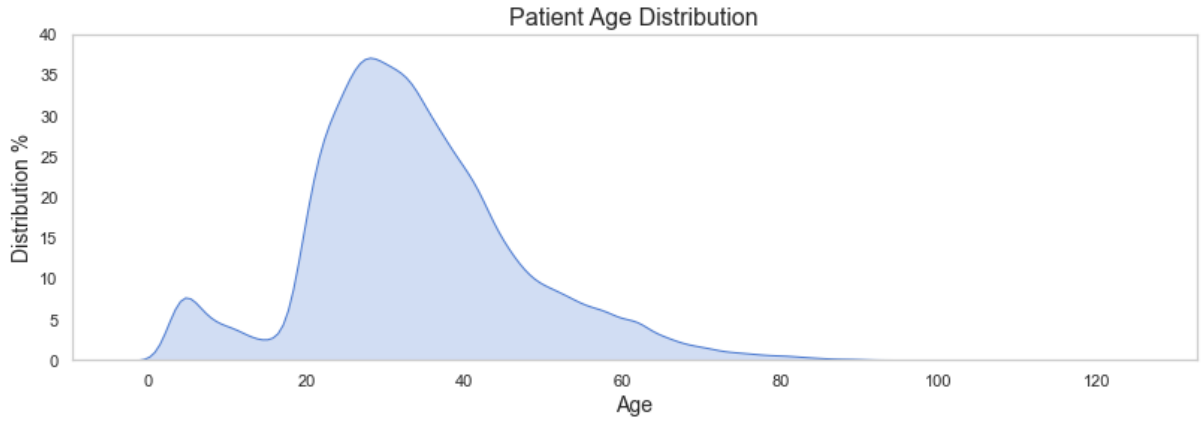


Fig. The age distribution of all patients ranges from 0 - 122 years. The distribution slightly resembles normal distribution shape. The mean age is approximately 35 years old. The most frequent age is approximately 30 years old.

7. Age Distribution of diabetic vs none-diabetic patients

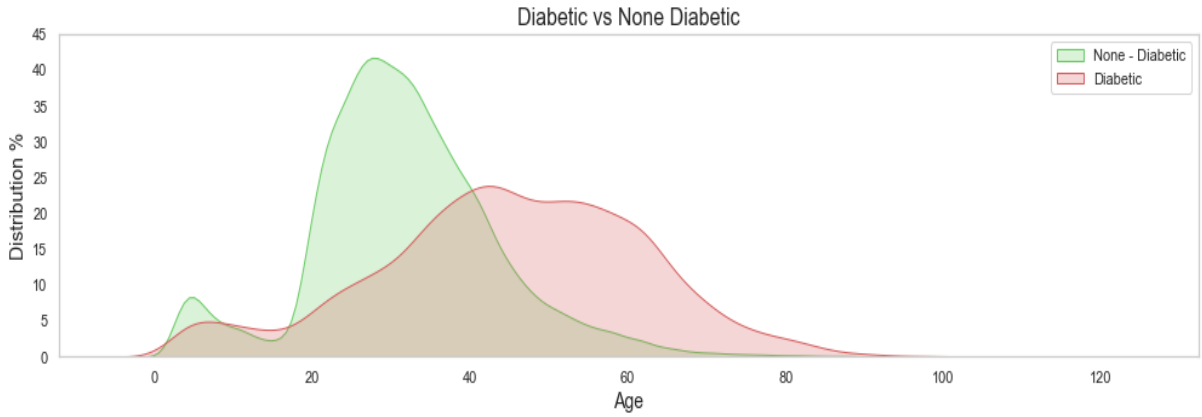


Fig. The majority of incidents of diabetes is between those aged 42 - 60 years old. The proportion of diabetic vs none-diabetic patients are somewhat equal around the age of 40 years old. Diabetic patients aged 0 - 18 years old are most likely type 1 diabetic patients (hereditary diabetes mellitus). The trough between the ages 14 - 18 could be due to infrequent hospital visits by this age group. The mean age for none-diabetic patients is approximately 32 years old.

8. Age distribution of diabetic and hypertensive patients

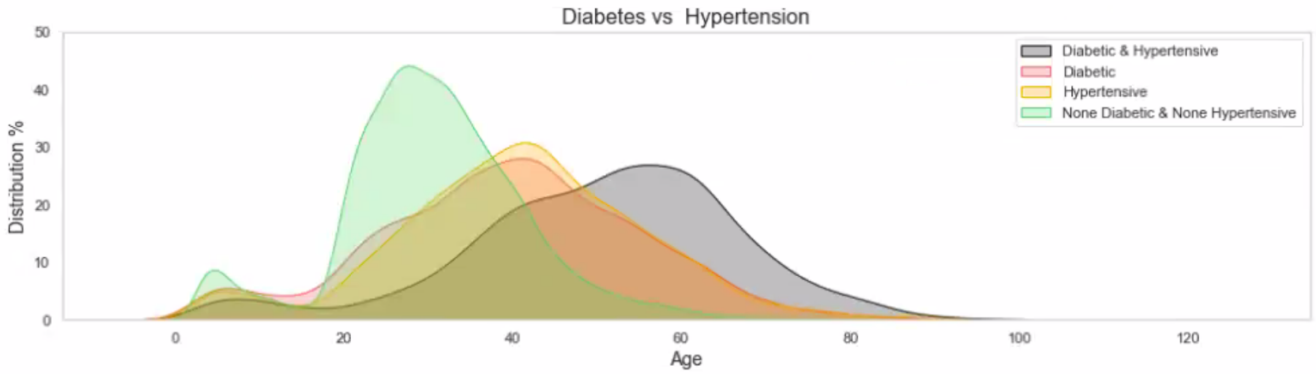


Fig. Age distribution of diabetic and hypertensive patients. The distribution pattern indicates younger individuals are at risk of developing hypertension and/or diabetes. Both diabetes and hypertension exhibit similar distribution patterns affecting the similar age groups (30 - 60 years old). Older individuals (50+ years old) are at high odds of developing both diabetes and hypertension.

Visualization: International data

1. **Diabetes patients by gender**

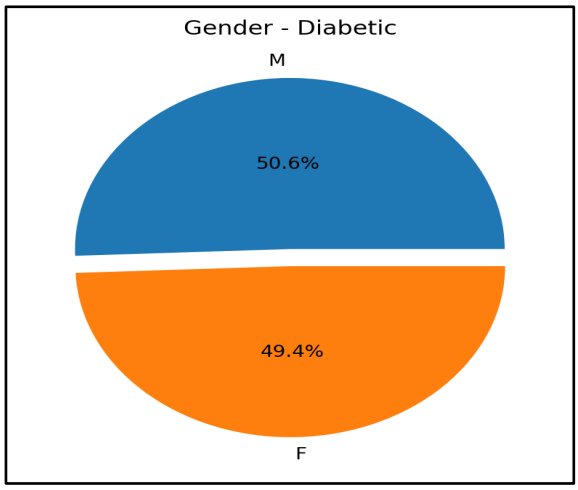


Fig. It seems that people with diabetes of both sexes have similar proportions

2. **Age distribution of diabetic and hypertensive patients – international data**

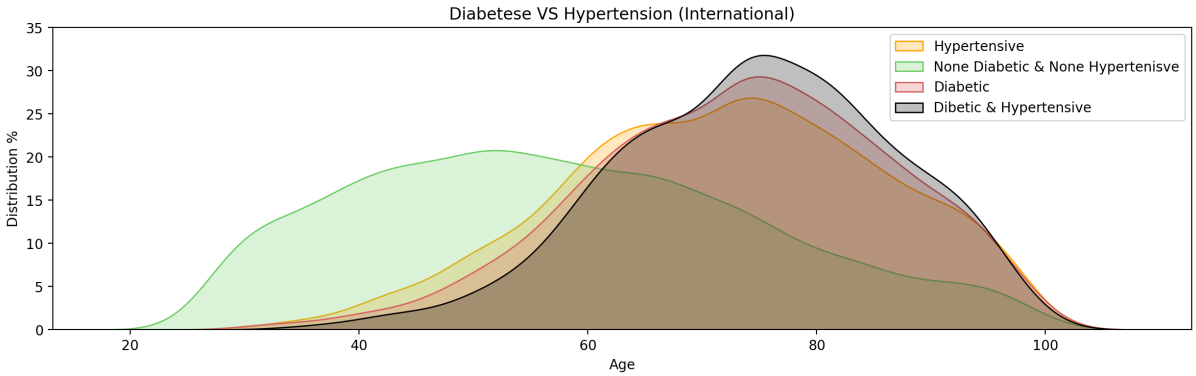


Fig. It seems that hypertension and diabetes are associated with the same age group, as the vast majority of sufferers fall. within the elderly, that is, over sixty years old, which means that the patients here are mostly from type 2 diabetes or that the data were collected only for adults.

Data Preparation

Goal: To enrich all predictors and improve model accuracy.

Initial data cleaning

- Checking nulls
- Remove duplicates
- Merge similar values
- Correcting unusual characters (?, !, etc)
- Remove or obtain absolute value for non-logical values (example, minus age/height).

Feature engineering

- Feature engineer BMI column from height & weight columns
- Diagnosis description (selected top 6 diabetic predictors based on domain knowledge)
- Feature engineer age from date of birth
- Address different filling strategies in important columns

    - Hypertension column:
        - 1. Fill nulls with false value (ie, patient does not have hypertension)
        - 2. Remove null values
    - Height & Weight columns:
        - 1. Obtain absolute values
        - 2. Determine outlier threshold values (drop values > 260 kg/cm)
        - 3. Nulls of age group 0-20 years old were filled with average reference values obtained from external source[5]
        - Since they contained none-logical values.
        - Nulls of age group >20 years old,
            - fill weight null values with median (gender specific)
            - fill height null values with mean (gender specific)
    - Date of birth column:
        - Fill null values with date majority (contained minor nulls) then feature engineer age column.

Cleaning and feature engineering scenarios

Different cleaning scenarios were applied along with specific modelling (**Table 5**)

Table 5. Cleaning and feature engineering scenarios

Scenario No	Age	Height	Weight	BMI	Has hypertension
1	Fill with mean	Fill with mean	Fill with mean	-	Fill with most frequent value
2	Fill with mean	Fill with mean (Gender specific)	Fill with mean (Gender specific	-	Fill with most frequent value
3	Fill with mean	drop	drop	-	Fill with most frequent value
4	Fill with mean	drop	drop	-	Drop null values
5.1	Fill with mode	drop	drop	BMI	Fill with most frequent value
5.2	Fill with mode	drop	drop	BMI	Drop null values

Dummies

- Transform (encode) categorical columns before modelling. Example columns include: Gender, Nationality, Hypertension, Has diabetes.

Modelling & Evaluation

Baseline Score

- Baseline score was computed according to the higher class in the target column
    - Local data target: has diabetes
    - International data target: DMIendecater

Data balancing

Initial data contained unbalanced classes, hence, we attempted to balance the classes prior to modelling via:

- Random sampling choice of (0, 1) in order to equalize the minority class (1) with the majority class (0)
- Over & under sampling
- Smote

Split

- Train data (70%)
- Test data (30%)

Feature scaling

- Standardize the training data (fit & transform)
- Apply standardized data on test data

Models instantiation

- Fit specified model on train data and predict test results

Evaluation metrics

- Evaluate predicted results with actual results (score)
- ROC-AUC score
- Confusion matrix (True positive, True negative, False positive & False negative)
- Classification report for train and test sets (Precision, recall & F1 score)

Results

Table 6: Cleaning, feature engineering scenarios and f1-score summary

Features	Scenario1	Scenario2	Scenario3	Scenario4	Scenario5.1	Scenario5.2
Height	Mean	Mean-GS				
Weight	Mean	Mean-GS				
BMI					*	*
Age	Mean	Mean	Mean	Mean	Mode	Mode
Hypertension						
Type-1						
Type-2						
Hyperglycaemia						
hypercholesterolaemia						
Obesity						
Saudi						
Female						
has-hypertension	MFV	MFV	MFV	DNV	MFV	DNV
F1- Score	0.74063	0.73968	0.7422	0.7474	0.7432	0.7228
Model used	Gradient Boosting Classifier	Cat Boost Classifier	Gradient Boosting Classifier	Cat Boost Classifier	Cat Boost Classifier	Cat Boost Classifier

Mean-GS: gender-specific mean
MFV: most frequent value (False, patient does not have hypertension)
DNV: drop null values
*: for filling strategies refer to data preparation - page 18

Color Key: Feature Used | Feature Not Used

International data

Basiline score

```
In [620]:
# compute baseline score DMIndicator == 0
international[international.DMIndicator == 0].shape[0]/international.shape[0]

Out[620]: 0.8086047446722959
```

Our target is not balanced, as our data is biased by 80% towards zero more than towards one. Hence, we will first attempt to make it balanced. In order to achieve this goal, we have taken several different methods including under sampling, over sampling with SMOTE, rebalancing of target class (randomly chosen) or select specific rows with majority class to be equal to the minority class to compare them later.

Model score is sometimes high but the impacts of imbalanced data are implicit, i.e., it does not raise an immediate error when building and running the model, but the results can be deceptive. If the degree of class imbalance for the majority class is extreme, then a machine trained classifier may yield high overall prediction accuracy since the model is most likely to predict most samples belonging to the majority class.

Random Undersampling

```
In [219]:
# summarize class distribution
print("Before undersampling: ", Counter(y_train))

# define undersampling strategy
undersample = RandomUnderSampler(sampling_strategy='majority')

# fit and apply the transform
Xs_train_under, y_train_under = undersample.fit_resample(Xs_train, y_train)

# summarize class distribution
print("After undersampling: ", Counter(y_train_under))

Before undersampling:  Counter({0: 5630, 1: 1333})
After undersampling:  Counter({0: 1333, 1: 1333})
```

Before applying undersampling technique, the target column (DMindicator) had unbalanced classes (0,1). Classes were balanced after application of undersampling (both 1,0 have the same number of rows).

Oversampling with SMOTE

In [223]:
```
# summarize class distribution
print("Before oversampling: ",Counter(y_train))

# define oversampling strategy
SMOTE = SMOTE()

# fit and apply the transform
Xs_train_SMOTE, y_train_SMOTE = SMOTE.fit_resample(Xs_train, y_train)

# summarize class distribution
print("After oversampling: ",Counter(y_train_SMOTE))
```

```
Before oversampling:  Counter({0: 5630, 1: 1333})
After oversampling:  Counter({0: 5630, 1: 5630})
```

Before applying oversampling technique, the target column (DMindicator) had unbalanced classes (0,1). Classes were balanced after application of oversampling (both 1,0 have the same number of rows). Please refer to the jupyter notebook for full details of scores.

Rebalance Target Class (Randomly Chosen)

Randomly Choice

In [231]:
```

# rebalancing target class [0, 1]
# select the same number of rows in case DMIndicator == 1 for case DMIndicator ==0
international_sample_class_1 = y_train[y_train == 1] # 5228
international_index1 = np.random.choice(y_train[y_train == 0].index, 1238, replace=False)
international_sample_class_0 = y_train[international_index1]

print(f'class 0 shape: {international_sample_class_0.shape}')
print(f'class 1 shape: {international_sample_class_1.shape}')



#Do Scaling for data
ss = StandardScaler()
Xs_train = ss.fit_transform(X_train)
Xs_test = ss.transform(X_test)
```

```
class 0 shape: (1238,)
class 1 shape: (1238,)
```

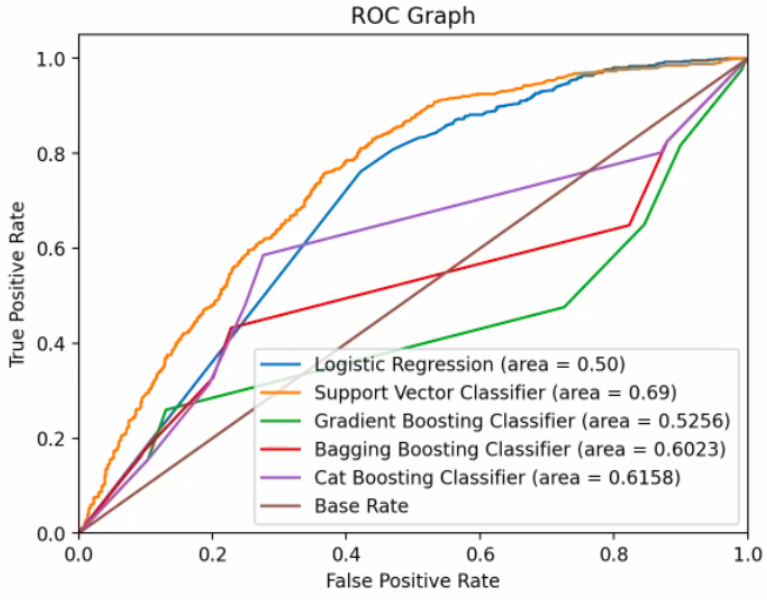
Before applying random choice technique, the target column (DMindicator) had unbalanced classes (0,1). Classes were balanced after application of random choice (both 1,0 have the same number of rows).

In [237]:
```
for model in models:
    f1_sc = applying_model(models[model], Xs_train, Xs_test, y_train, y_test)
    print(f'F1-Score {model} : {f1_sc}')
```

```
F1-Score Logistic Regression : 0.698016164584864
F1-Score Gradient Boosting Classifier : 0.7159090909090908
F1-Score Bagging Classifier  : 0.6333333333333333
```

```
F1-Score Cat Boost Classifier : 0.7090395480225988
F1-Score SVC : 0.7159479808350445
```

Best f-1 score was obtained using the Support Vector Classifier.



Based on f1-score, the AUC score should return the best AUC which was obtained using the Support Vector Classifier.

Local data

Basiline score

```
In [239]: scenario_1[scenario_1['diabetetic'] == 0].shape[0]/scenario_1.shape[0]
Out[239]: 0.8510947357260245
```

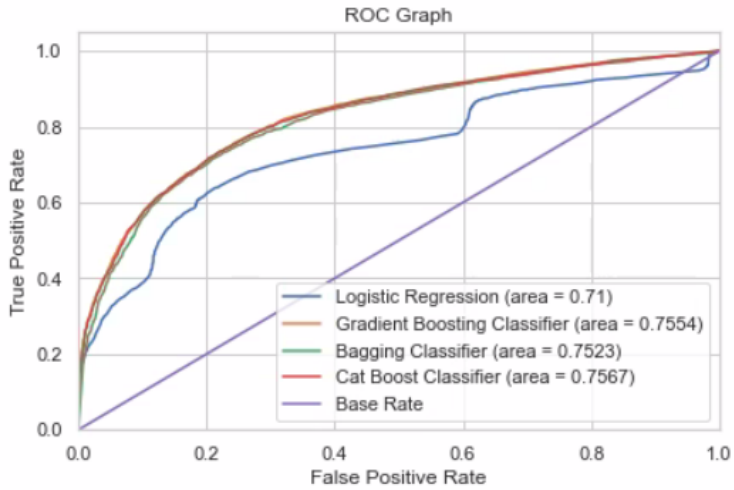
```
In [270]: # select the same number of rows in case DMIndicator == 1 for case DMIndicator ==0
          scenario_4_sample_class_1 = scenario_4[scenario_4['diabetetic'] == 1]
          scenario_4_sample_class_0 = scenario_4[scenario_4['diabetetic'] == 0][:11801]


          df = scenario_4_sample_class_1.append(scenario_4_sample_class_0)
```

Before applying random choice technique, the target column (diabetic) had unbalanced classes (0,1). Classes were balanced after application of random choice (both 1,0 have the same number of rows).

```
F1-ScoreLogistic Regression: 0.6780665140702842
F1-ScoreGradient Boosting Classifier: 0.7460736273401182
F1-ScoreBagging Classifier : 0.7459230673471927
```

```
F1-ScoreCat Boost Classifier: 0.7474239758733351
```

The best f-1 score was obtained using the Cat Boost Classifier.



Based on f1-score, the AUC score should return the best AUC which was obtained using the Cat Boost Classifier.

Conclusion

International data

Overall, this dataset was clean and easy to handle. However, the challenge was dealing with unbalanced data without affecting the integrity of the medical data.

General insights from the data show a tendency of diabetes to affect both genders equally. Similar to local observations, hypertension is the main diabetes comorbidity affecting almost the same age group and with a disease onset of 40 years old and above.

With regards to modelling, three different data balancing methods were employed including over sampling, under sampling and manual resampling. The highest score was achieved by using the Support Vector Machine Classifier (SVC) and the Gradient Boosting Classifier (f1-score: 71%).

Local data

To summarize, we can say that height and weight are very important factors to improve the accuracy of the model, as the scenarios that were tested and included these two characteristics gave us better results reaching as hight as 74%. Hence, we recommend scenario_4 as it is more reliable and closely resembles the original data (has fewer cleaning processes). As such, we can depend on the model used in scenario_4 (Cat Boost Classifier) for predicting future diabetic individuals.

Also, the percentage of diabetes in the local data (Saudi population) is 14%, and 70.9% of which are males. Additionally, these male diabetic patients mostly report pain complications.

Recommendations and concluding remarks

The model accuracy could be improved by testing different balancing / resampling techniques such as under sampling with over sampling along with SOMTE

It is also recommended to consult a medical data scientist (bioinformatician) expert regarding best practice when handling medical data.

As a team, we recommend that we first improve data entry methods and be more mindful about its integrity, as many columns were discarded despite their great importance as they contained many missing data which may have otherwise drastically improved our model.

Examples of missing values that may have been valuable predictors for our model include the lab results for patients and steps per day.

In addition to the missing data, there are a lot of columns with illogical or false values, such as the marital status column. Also, we recommend collecting different data or features such as family history as they can be important, especially in the case of type 1 diabetes.

Furthermore, providing family history will aid in analysing trends and patterns of several years. Therefore, it is safe to assume that the quality of data would highly improve our model score and accuracy.

REFERENCES

1. Kharroubi, A. T. Diabetes mellitus: The epidemic of the century. *World J. Diabetes* **6**, 850 (2015).
2. Ndisang, J. F., Vannacci, A. & Rastogi, S. Insulin Resistance, Type 1 and Type 2 Diabetes, and Related Complications 2017. *Journal of Diabetes Research* vol. 2017 (2017).
3. Diabetes - Statistics & Facts | Statista. https://www.statista.com/topics/1723/diabetes/.
4. Abdulaziz Al Dawish, M. *et al.* Diabetes Mellitus in Saudi Arabia: A Review of the Recent Literature. *Curr. Diabetes Rev.* **12**, 359–368 (2016).
5. Average Height to Weight Chart: Babies to Teenagers | Disabled World. https://www.disabled-world.com/calculators-charts/height-weight-teens.php.