

# Diabetes in Youth and Adults in India

The Project aims at finding the number of youth and adults suffering from diabetes based on various parameters such as Age group, Alcohol consumption, Gender, Genetic Risk Score and The type of diabetes they are suffering from according to the age group.

The whole data set is divided into two age groups, i.e., 15-20 and 20-25. The aim of the project is to analyse how these all are related to the nature of diabetes people are suffering from.

Vanshik Narula

[vanshiknarula03@gmail.com](mailto:vanshiknarula03@gmail.com)

---

# DIABETES IN YOUTH AND ADULTS IN INDIA ANALYSIS

---

Step 1: Import all the libraries

Code:

```
1 import pandas as pd
2 import numpy as np
3 import matplotlib.pyplot as plt
4 import seaborn as sns
```

Step 2: Read the Dataset

Code & Output:

```
1 df=pd.read_csv("C:\\Users\\MY PC\\Downloads\\diabetes_young_adults_india.csv")
```

```
1 df.head()
```

	ID	Age	Gender	Region	Family_Income	Family_History_Diabetes	Parent_Diabetes_Type	Genetic_Risk_Score	BMI	Physical_Activity_Level	...	Smoking
0	1	21	Male	North	2209393	No	None	6	31.4	Sedentary	...	Yes
1	2	18	Female	Central	387650	No	None	5	24.4	Active	...	No
2	3	25	Male	North	383333	No	None	6	20.0	Moderate	...	No
3	4	22	Male	Northeast	2443733	No	None	4	39.8	Moderate	...	No
4	5	19	Male	Central	1449463	No	None	4	19.2	Moderate	...	No

5 rows × 22 columns



### Step 3: Checking Data Info

#### Code & Output:

```
1 df.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 100000 entries, 0 to 99999
Data columns (total 22 columns):
#   Column                               Non-Null Count  Dtype
---  -
0   ID                                   100000 non-null  int64
1   Age                                   100000 non-null  int64
2   Gender                               100000 non-null  object
3   Region                               100000 non-null  object
4   Family_Income                        100000 non-null  int64
5   Family_History_Diabetes              100000 non-null  object
6   Parent_Diabetes_Type                 100000 non-null  object
7   Genetic_Risk_Score                   100000 non-null  int64
8   BMI                                   100000 non-null  float64
9   Physical_Activity_Level               100000 non-null  object
10  Dietary_Habits                        100000 non-null  object
11  Fast_Food_Intake                      100000 non-null  int64
12  Smoking                               100000 non-null  object
13  Alcohol_Consumption                   100000 non-null  object
14  Fasting_Blood_Sugar                   100000 non-null  float64
15  HbA1c                                  100000 non-null  float64
16  Cholesterol_Level                     100000 non-null  float64
17  Prediabetes                           100000 non-null  object
18  Diabetes_Type                         100000 non-null  object
19  Sleep_Hours                           100000 non-null  float64
20  Stress_Level                           100000 non-null  int64
21  Screen_Time                           100000 non-null  float64
dtypes: float64(6), int64(6), object(10)
memory usage: 16.8+ MB
```

### Step 4: Checking the null values

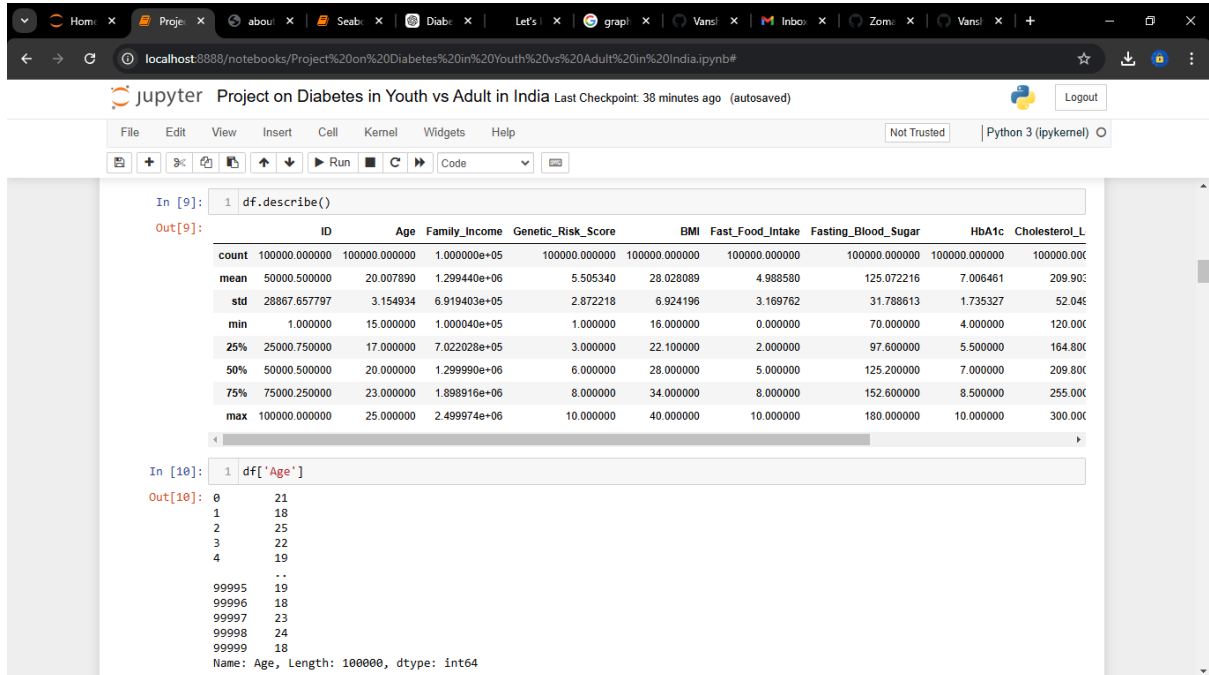
#### Code & Output:

```
1 df.isna().sum()

ID                0
Age               0
Gender            0
Region            0
Family_Income     0
Family_History_Diabetes  0
Parent_Diabetes_Type  0
Genetic_Risk_Score  0
BMI               0
Physical_Activity_Level  0
Dietary_Habits    0
Fast_Food_Intake  0
Smoking           0
Alcohol_Consumption  0
Fasting_Blood_Sugar  0
HbA1c             0
Cholesterol_Level  0
Prediabetes       0
Diabetes_Type     0
Sleep_Hours       0
Stress_Level      0
Screen_Time       0
dtype: int64
```

## Step 5: Descriptive Analysis Of dataset

### Code & Output:



```
In [9]: 1 df.describe()
Out[9]:
```

	ID	Age	Family_Income	Genetic_Risk_Score	BMI	Fast_Food_Intake	Fasting_Blood_Sugar	HbA1c	Cholesterol_L
count	100000.000000	100000.000000	1.000000e+05	100000.000000	100000.000000	100000.000000	100000.000000	100000.000000	100000.000000
mean	50000.500000	20.007890	1.299440e+06	5.505340	28.028089	4.988580	125.072216	7.006461	209.900000
std	28867.657797	3.154934	6.919403e+05	2.872218	6.924196	3.169762	31.788613	1.735327	52.040000
min	1.000000	15.000000	1.000040e+05	1.000000	16.000000	0.000000	70.000000	4.000000	120.000000
25%	25000.750000	17.000000	7.022028e+05	3.000000	22.100000	2.000000	97.600000	5.500000	164.800000
50%	50000.500000	20.000000	1.299990e+06	6.000000	28.000000	5.000000	125.200000	7.000000	209.800000
75%	75000.250000	23.000000	1.898916e+06	8.000000	34.000000	8.000000	152.600000	8.500000	255.000000
max	100000.000000	25.000000	2.499974e+06	10.000000	40.000000	10.000000	180.000000	10.000000	300.000000

```
In [10]: 1 df['Age']
Out[10]:
```

0	21
1	18
2	25
3	22
4	19
...	...
99995	19
99996	18
99997	23
99998	24
99999	18

Name: Age, Length: 100000, dtype: int64

## Step 6: Making a Column of Age groups:

### Code & Output:

```
1 bins = [15,20,26] # Define the bin edges
2 labels = ['15-20', '20-26']
```

```
1 grouped = df.groupby('Age_Group')
2
3 age_group_summary = grouped.size()
4
5 print(age_group_summary)
```

```
Age_Group
15-20    45258
20-26    54742
dtype: int64
```

Step 7: Converting diabetes\_type column in 0,1 for further analysis

Code & Output:

```
1 # Filter data for diabetic patients
2 diabetic_data = df[df['Diabetes_Type'] != 'None']

df['Has_Diabetes'] = df['Diabetes_Type'].apply(lambda x: 1 if x != 'None' else 0)
df.head()
```

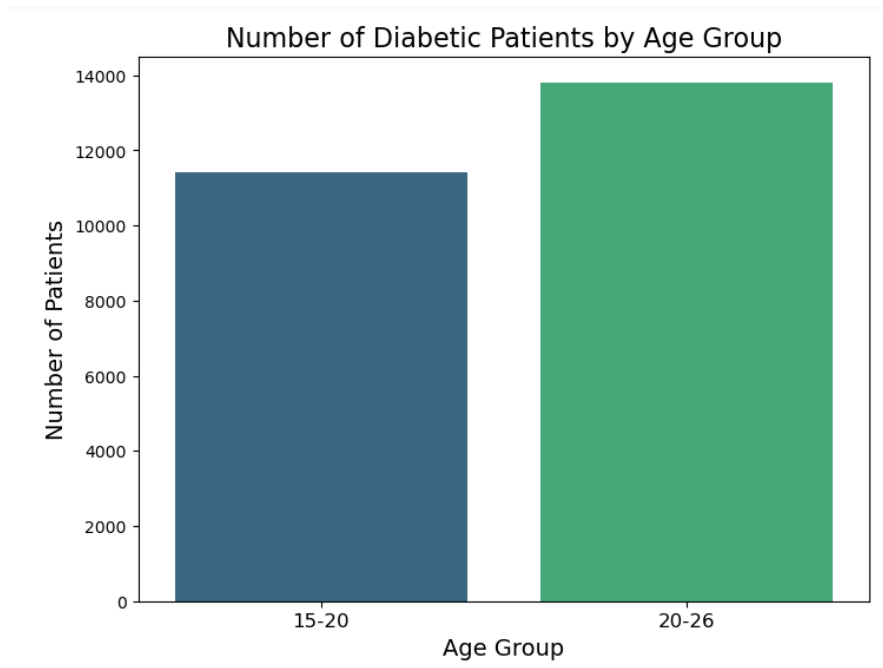
	ID	Age	Gender	Region	Family_Income	Family_History_Diabetes	Parent_Diabetes_Type	Genetic_Risk_Score	BMI	Physical_Activity_Level	...	Fasting_BI
0	1	21	Male	North	2209393	No	None	6	31.4	Sedentary	...	
1	2	18	Female	Central	387650	No	None	5	24.4	Active	...	
2	3	25	Male	North	383333	No	None	6	20.0	Moderate	...	
3	4	22	Male	Northeast	2443733	No	None	4	39.8	Moderate	...	
4	5	19	Male	Central	1449463	No	None	4	19.2	Moderate	...	

5 rows x 24 columns

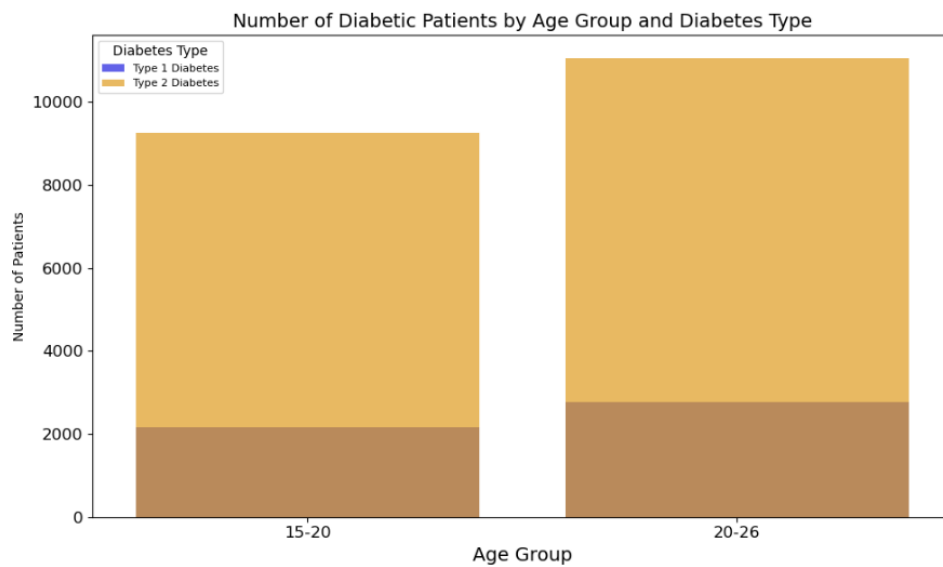
Cleaning is Completed

# DATA VISUALISATION

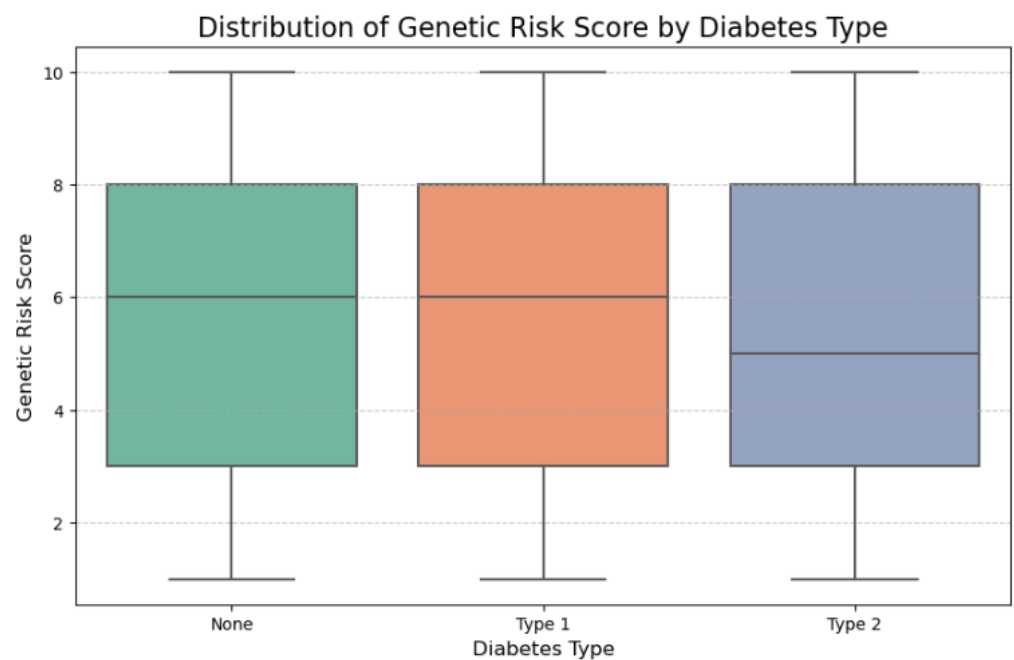
## 1. Age Group Wise Diabetic Patients.



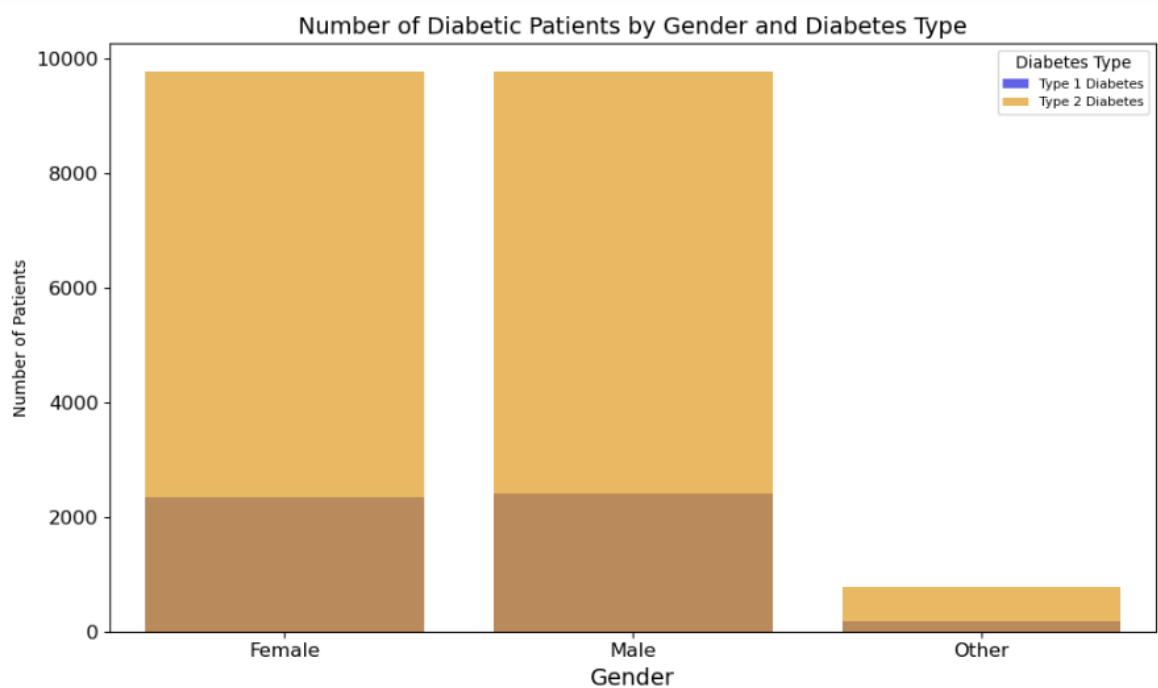
## 2. Age Group Wise Diabetic Type of Patients



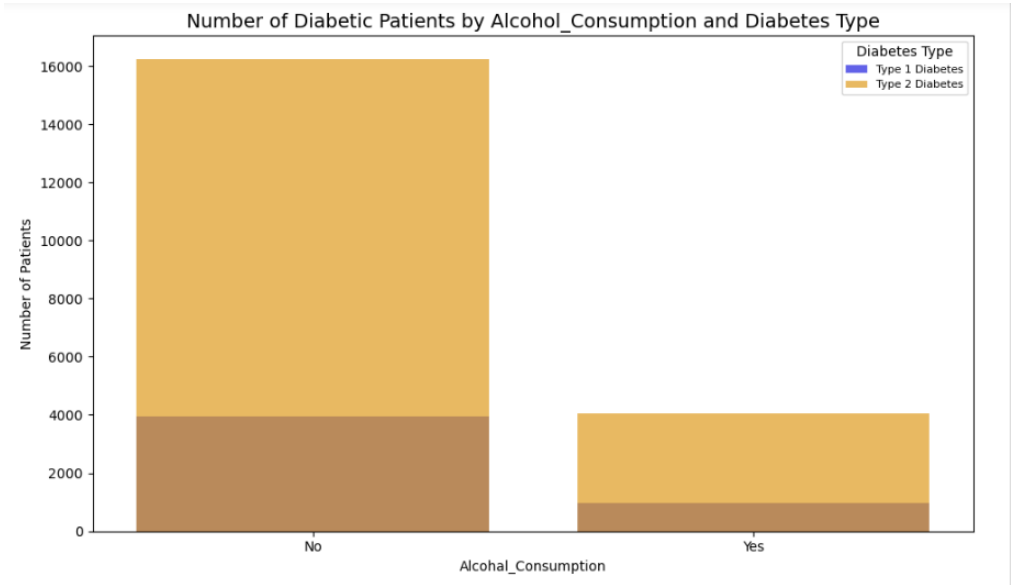
3. Boxplot for Comparison of Genetic Risk Score with Diabetes Type:



4. Distribution of Diabetic Patients With respect to Gender



## 5. Diabetic Patients with respect to alcohol consumption



# DASHBOARD

## Diabetes in youth and Adult

