

## **AI- BASED DIABETES PREDICTION SYSTEM**

ai-phase5

### Problem Statement:

The code seems to be related to a dataset that appears to be about diabetes. However, the problem statement is not explicitly mentioned in the code. To formulate a problem statement, you should clarify what you intend to achieve or predict with this dataset. For example, it could be predicting whether an individual has diabetes (Outcome = 1) based on various features, or it could be some other data analysis or prediction task related to diabetes.

### Design Thinking Process:

Understand the Problem: Clearly define the problem or objective you want to address with this dataset. In this case, it seems to be related to diabetes data, but the specific problem is not stated.

Data Collection: You've loaded the data from a CSV file named "diabetes.csv." It would be important to describe the source and context of the dataset if available.

Data Preprocessing: Your code shows some initial data preprocessing steps, such as loading the data, examining data types, and checking for missing values. It's important to clean and preprocess the data to ensure its quality for analysis.

Data Visualization: You've created various data visualizations (pie chart, bar graph, and histogram) to explore the data and understand its distribution, particularly with respect to age.

#### Development Phases:

From the code provided, it's clear that the development process involves the following phases:

Data Loading and Inspection: Loading the dataset using Pandas and inspecting it to understand its structure.

Data Preprocessing: Handling missing values, checking data types, and selecting specific columns of interest (e.g., 'Pregnancies', 'BMI', 'Age', 'Insulin').

Data Visualization: Creating various plots (pie chart, bar graph, and histogram) to visually explore the data's distribution, particularly focusing on the 'Age' column.

Analysis: You need to analyze the data to answer specific questions or address a particular problem. This step is missing from the provided code.

## Choice of Machine Learning Algorithm:

Your code does not mention the use of machine learning algorithms, model training, or evaluation metrics. If your goal is to build a predictive model, you would need to select a suitable machine learning algorithm (e.g., logistic regression, random forest, SVM) and define appropriate evaluation metrics (e.g., accuracy, precision, recall, F1-score) to assess the model's performance.

## Feature Extraction Techniques:

Feature extraction is not explicitly discussed in the provided code. It's important to select relevant features for your analysis or modeling. In your code, you have chosen a subset of columns for analysis.

## Innovative Techniques:

Your code demonstrates common data analysis and visualization techniques. However, it does not appear to use innovative or advanced techniques. Innovative techniques might involve advanced machine learning models, feature engineering, or

0	6	6	33.6	50	0
1	1	1	26.6	31	0
2	8	8	23.3	32	0
3	1	1	28.1	21	94

unique data preprocessing methods.

November 1, 2023

```
[3]: # importing the required python libraries
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
import plotly.express as px
%matplotlib inline
```

```
[4]: import pandas as pd
df=pd.read_csv("D:\calis\diabetes.csv")
df.head()
```

```
[4]: Pregnancies  Glucose  BloodPressure  SkinThickness  Insulin   BMI  \
0             6      148             72             35         0  33.6
1             1       85             66             29         0  26.6
2             8      183             64              0         0  23.3
3             1       89             66             23        94  28.1
4             0      137             40             35       168  43.1
```

```
DiabetesPedigreeFunction  Age  Outcome
0      0.627 50      1 1    0.351 31      0 2
      0.672 32      1 3    0.167 21      0
4      2.288 33      1
```

```
[6]: pr=df[['Pregnancies','Pregnancies','BMI','Age','Insulin']]
pr.head(4)
```

```
[6]: Pregnancies  Pregnancies  BMI  Age  Insulin
```

```
[19]: Pregnancies
```

```
0      111
1      135
2      103
3       75
4       68
5       57
6       50
7       45
8       38
9       28
10      24
```

```
[19]: df.groupby("Pregnancies").size()
11      11
```

```
12      9
13     10
14      2
15      1
17      1
dtype: int64
```

```
[7]: pr.groupby('Age').size()
```

```
[7]: Age
```

```
21     63
22     72
23     38
24     46
25     48
26     33
27     32
28     35
29     29
30     21
31     24
32     16
33     17
34     14
35     10
36     16
37     19
38     16
39     12
40     13
41     22
42     18
43     13
```

```

44      8
45     15
46     13
47      6
48      5
49      5
50      8
51      8
52      8
53      5
54      6
55      4
56      3
57      5
58      7
59      3
60      5
61      2
62      4
63      4
64      1
65      3
66      4
67      3
68      1
69      2
70      1
72      1
81      1
dtype: int64

```

```
[8]: df.info()
```

```

<class
'pandas.core.frame.DataFrame'>
RangeIndex: 768 entries, 0 to
767 Data columns (total 9
columns):
#   Column                Non-Null Count  Dtype
---  -
0   Pregnancies    768 non-null    int64
1   Glucose    768 non-null    int64
2   BloodPressure    768 non-null    int64

```

```

3  SkinThickness      768 non-null    int64
4  Insulin 768 non-null    int64
5  BMI      768 non-null    float64
6  DiabetesPedigreeFunction 768 non-null
   float64
7  Age      768 non-null    int64
8  Outcome 768 non-null    int64

```

```
dtypes: float64(2), int64(7)
```

```
memory usage: 54.1 KB
```

```
[11]: pr.isnull()
```

```

[11]:  Pregnancies  Pregnancies  BMI  Age  Insulin
      0          False      False False False  False
      1          False      False False False  False
      2          False      False False False  False
      3          False      False False False  False
      4          False      False False False  False
      ..          ...      ...   ...   ...   ...
      763         False      False False False  False
      764         False      False False False  False
      765         False      False False False  False
      766         False      False False False  False
      767         False      False False False  False
      [768 rows x 5 columns]

```

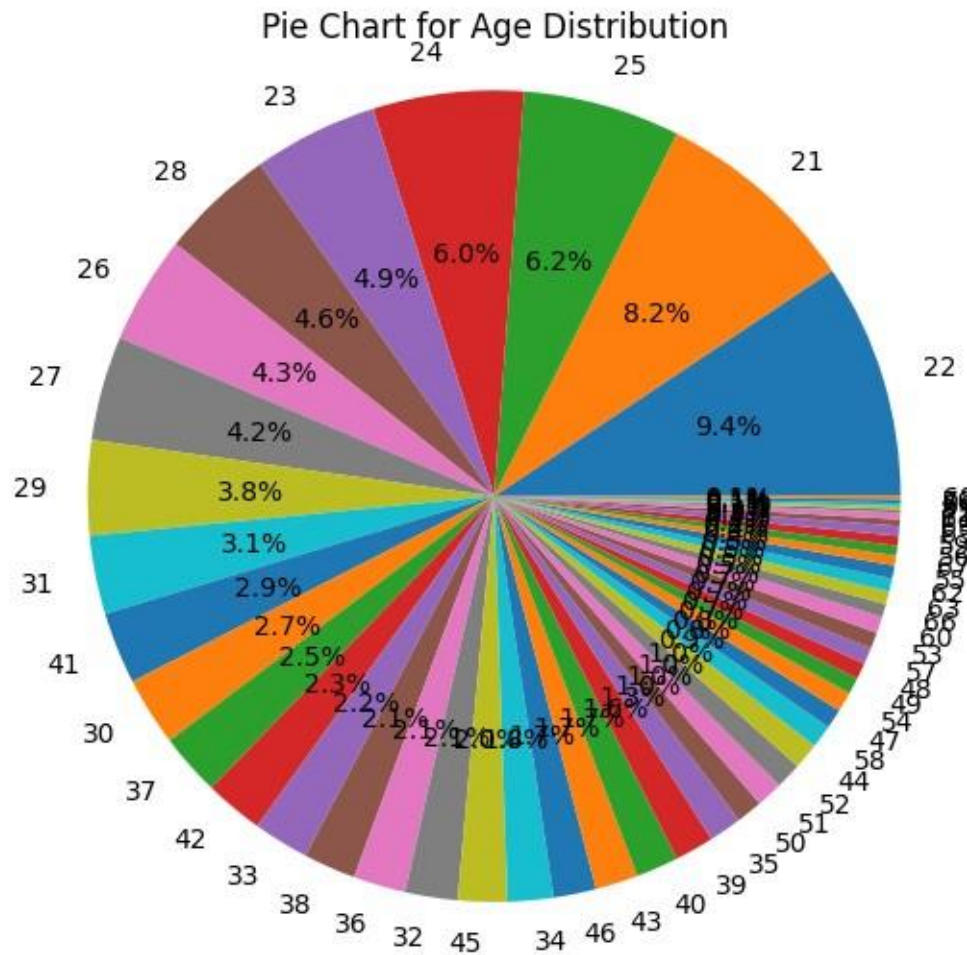
```

[15]: # Create a pie chart for the "Age" column
age_counts = df['Age'].value_counts()
labels = age_counts.index
sizes = age_counts.values

plt.figure(figsize=(6,6))
plt.pie(sizes, labels=labels, autopct='%1.1f%%')
plt.title("Pie Chart for Age Distribution")
plt.axis('equal') # Equal aspect ratio ensures that the pie chart is circular.

plt.show()

```

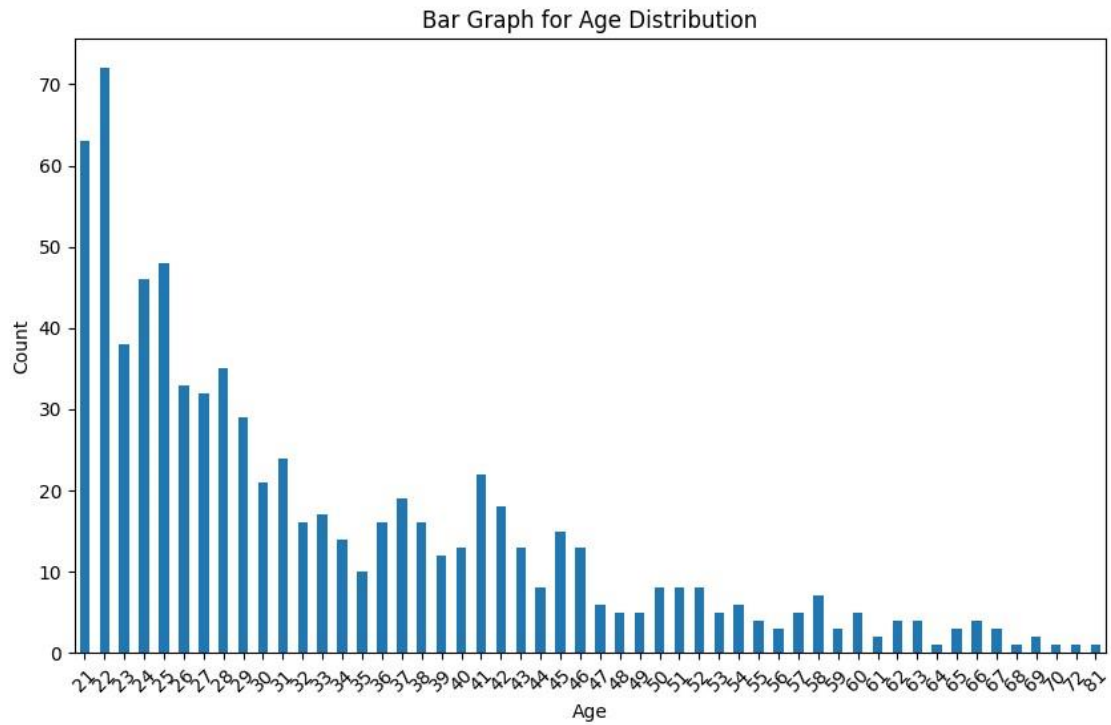


```
[17]: # Create a bar graph for the "Age" column
age_counts = df['Age'].value_counts().sort_index()

plt.figure(figsize=(10, 6))
age_counts.plot(kind='bar')
plt.title("Bar Graph for Age Distribution ")
plt.xlabel("Age")
plt.ylabel("Count")
plt.xticks(rotation=45)

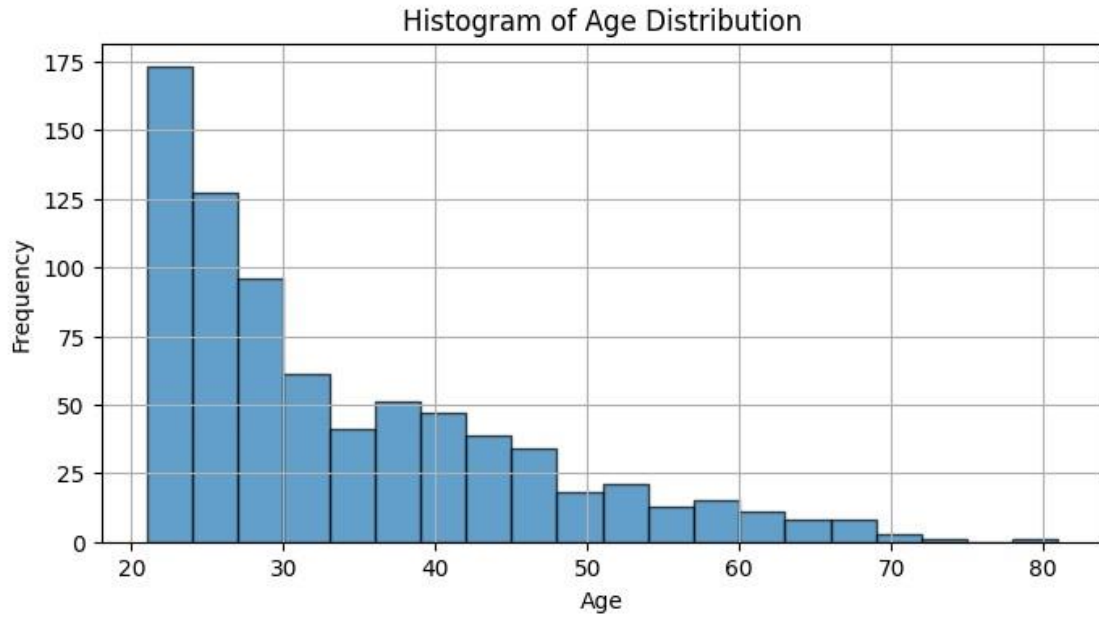
plt.show()
```





```
[21]: # Create a histogram for the "Age" column
plt.figure(figsize=(8, 4))
plt.hist(df['Age'], bins=20, edgecolor='k', alpha=0.7)
plt.title("Histogram of Age Distribution")
plt.xlabel("Age")
plt.ylabel("Frequency")
plt.grid(True)

plt.show()
```



### CONCLUSION:

You've provided code and visualizations related to a diabetes dataset, but a clear problem statement and details about the choice of machine learning algorithms and feature extraction techniques are missing. To provide a comprehensive overview, you should clearly define the problem, specify the machine learning approach, and describe the feature engineering and model evaluation processes.