Vidyavardhini’s College of Engineering & Technology

Abhishek Jani
BE - IT
Roll no - 50

Experiment No.1
Collect, Clean, Integrate and Transform Healthcare Data based on specific disease
Date of Performance: 4/8/2023
Date of Submission: 11/8/2023

Aim: Collect, Clean, Integrate and Transform Healthcare Data based on specific disease

Objective: The objective of this experiment is to perform basic pre processing on healthcare data set using python libraries

Theory:

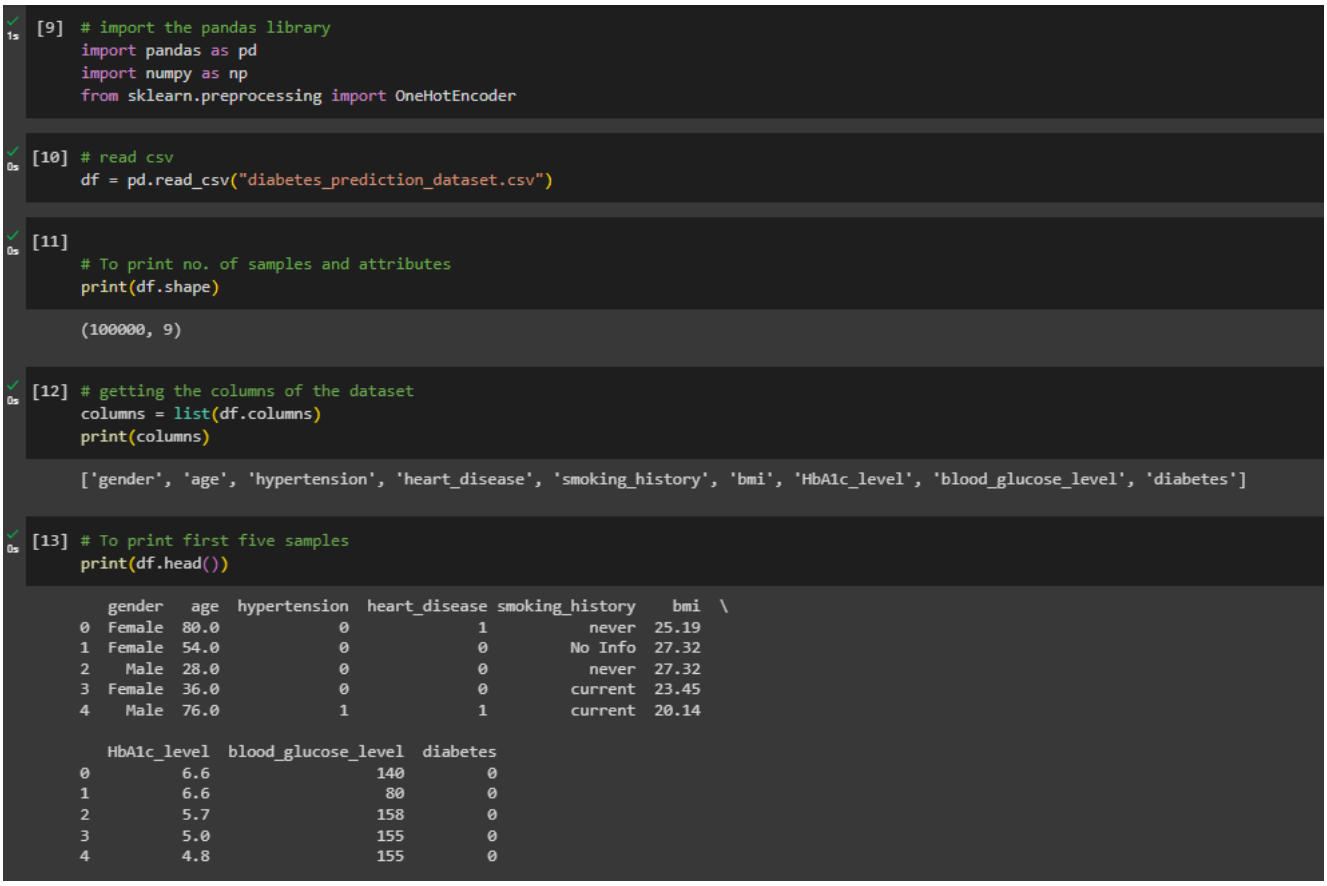
Data Collection- Data collection is the process of gathering and measuring information from countless different sources. In order to use the data we collect to develop practical artificial intelligence (AI) and machine learning solutions, it must be collected and stored in a way that makes sense for the business problem at hand.

Data Cleaning: Cleaning data refers to the way of deleting wrong, corrupted, wrongly formatted, duplicate information, or incomplete information from a dataset. The possibility of duplicating or mislabelling data increases when two or more data sources are combined.

Data Integration: Data integration is the practice of consolidating data from disparate sources into a single dataset with the ultimate goal of providing users with consistent access and delivery of data across the spectrum of subjects and structure types, and to meet the information needs of all applications and business processes.

Data transformation: Data transformation is the process of converting, cleansing, and structuring data into a usable format that can be analyzed to support decision making processes, and to propel the growth of an organization. Data transformation is used when data needs to be converted to match that of the destination system.

Code: -



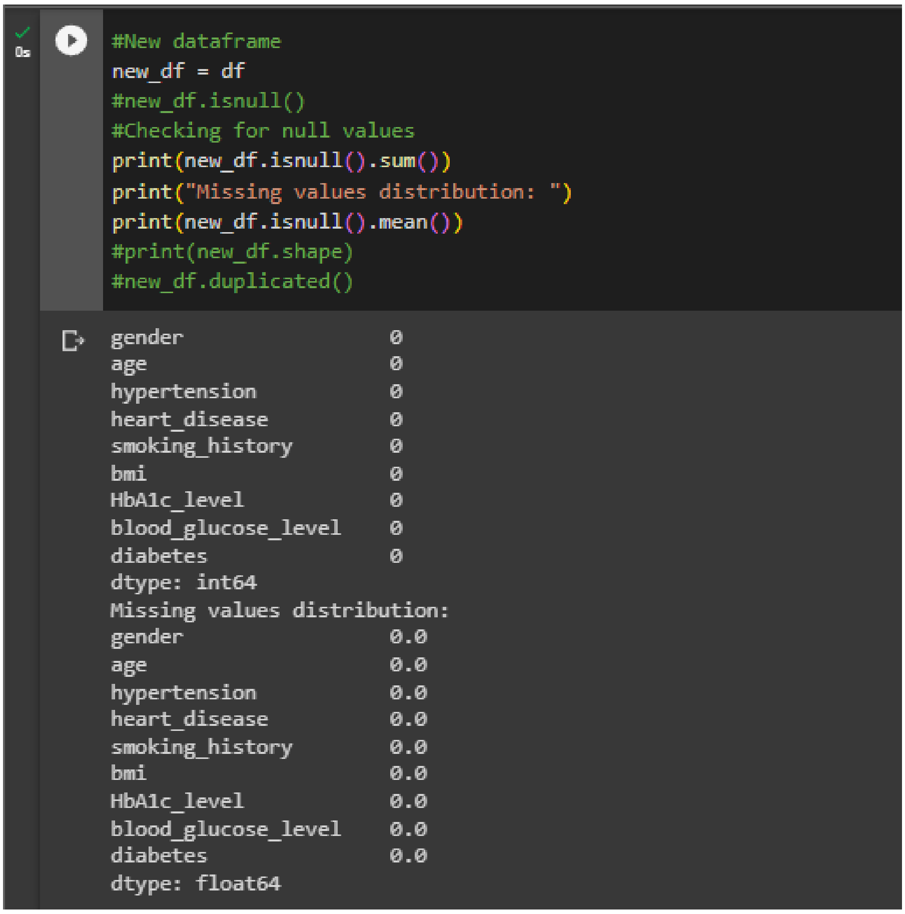
```
[9]  # import the pandas library
     import pandas as pd
     import numpy as np
     from sklearn.preprocessing import OneHotEncoder
```

```
[10] # read csv
     df = pd.read_csv("diabetes_prediction_dataset.csv")
```

```
[11]
     # To print no. of samples and attributes
     print(df.shape)

     (100000, 9)
```

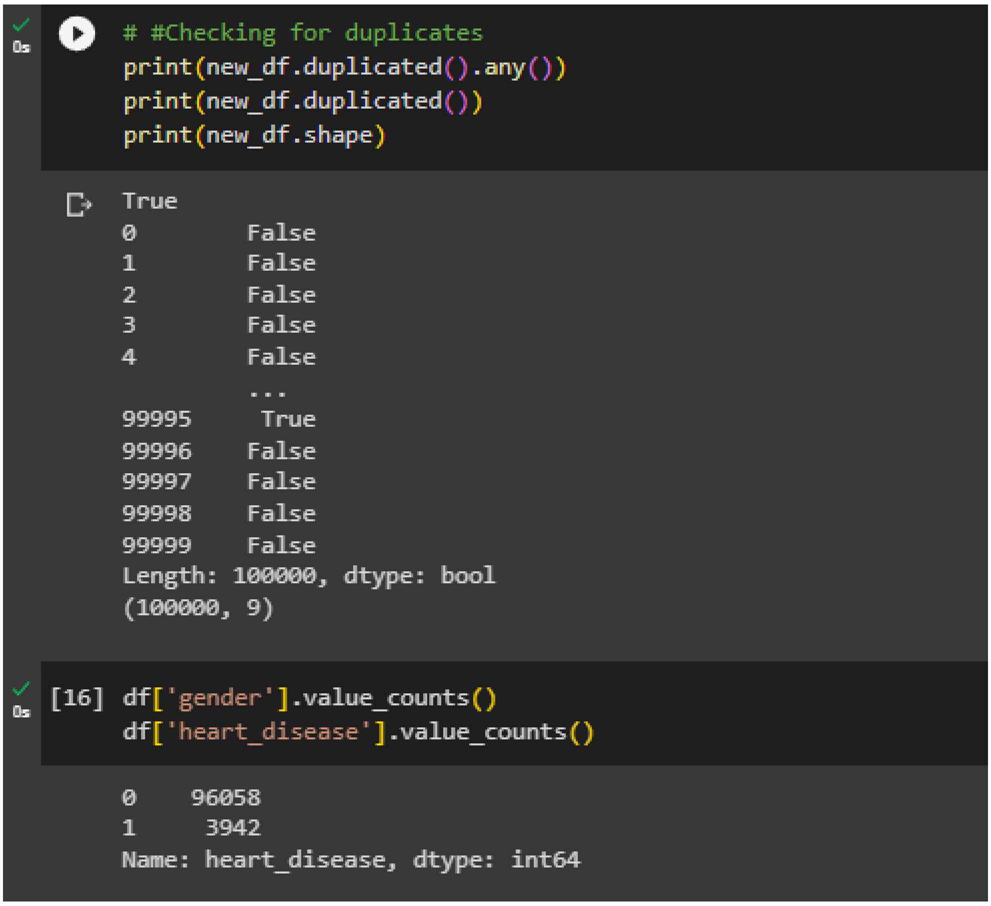
```
[12] # getting the columns of the dataset
     columns = list(df.columns)
     print(columns)

     ['gender', 'age', 'hypertension', 'heart_disease', 'smoking_history', 'bmi', 'HbA1c_level', 'blood_glucose_level', 'diabetes']
```

```
[13] # To print first five samples
     print(df.head())

        gender   age  hypertension  heart_disease smoking_history    bmi  \
     0  Female  80.0             0              1           never  25.19
     1  Female  54.0             0              0         No Info  27.32
     2    Male  28.0             0              0           never  27.32
     3  Female  36.0             0              0         current  23.45
     4    Male  76.0             1              1         current  20.14

        HbA1c_level  blood_glucose_level  diabetes
     0          6.6                  140         0
     1          6.6                   80         0
     2          5.7                  158         0
     3          5.0                  155         0
     4          4.8                  155         0
```



```
#New dataframe
new_df = df
#new_df.isnull()
#Checking for null values
print(new_df.isnull().sum())
print("Missing values distribution: ")
print(new_df.isnull().mean())
#print(new_df.shape)
#new_df.duplicated()
```

```
gender                 0
age                    0
hypertension           0
heart_disease          0
smoking_history        0
bmi                    0
HbA1c_level            0
blood_glucose_level    0
diabetes               0
dtype: int64
Missing values distribution: 
gender                 0.0
age                    0.0
hypertension           0.0
heart_disease          0.0
smoking_history        0.0
bmi                    0.0
HbA1c_level            0.0
blood_glucose_level    0.0
diabetes               0.0
dtype: float64
```

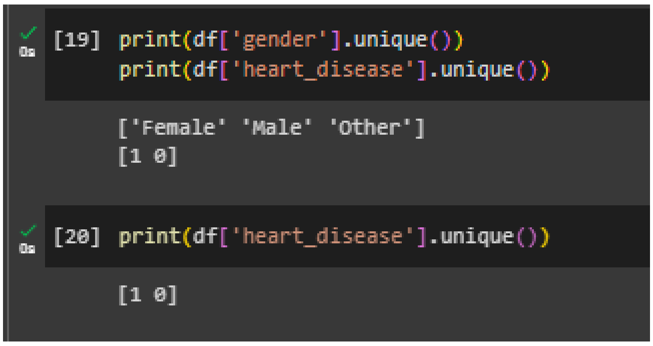


```
# #Checking for duplicates
print(new_df.duplicated().any())
print(new_df.duplicated())
print(new_df.shape)
```

```
True
0        False
1        False
2        False
3        False
4        False
         ...  
99995     True
99996    False
99997    False
99998    False
99999    False
Length: 100000, dtype: bool
(100000, 9)
```

```
[16] df['gender'].value_counts()
     df['heart_disease'].value_counts()
```

```
0    96058
1     3942
Name: heart_disease, dtype: int64
```

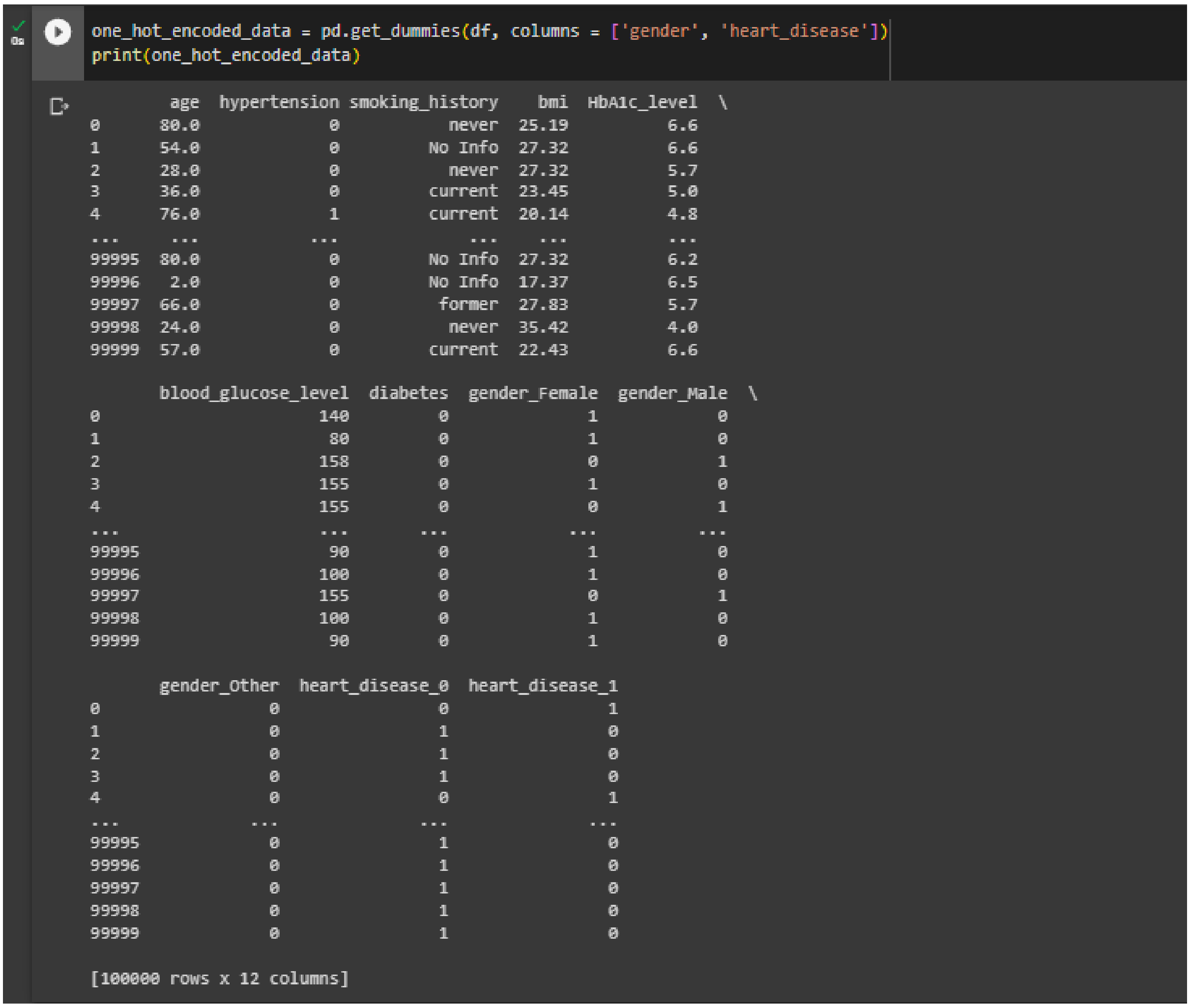


```
[19] print(df['gender'].unique())
     print(df['heart_disease'].unique())
```

```
['Female' 'Male' 'Other']
[1 0]
```

```
[20] print(df['heart_disease'].unique())
```

```
[1 0]
```



```
one_hot_encoded_data = pd.get_dummies(df, columns = ['gender', 'heart_disease'])
print(one_hot_encoded_data)
```

```
        age  hypertension smoking_history    bmi  HbA1c_level  \
0      80.0             0           never  25.19          6.6
1      54.0             0         No Info  27.32          6.6
2      28.0             0           never  27.32          5.7
3      36.0             0         current  23.45          5.0
4      76.0             1         current  20.14          4.8
...     ...           ...             ...    ...          ...
99995  80.0             0         No Info  27.32          6.2
99996   2.0             0         No Info  17.37          6.5
99997  66.0             0          former  27.83          5.7
99998  24.0             0           never  35.42          4.0
99999  57.0             0         current  22.43          6.6

       blood_glucose_level  diabetes  gender_Female  gender_Male  \
0                      140         0              1            0
1                       80         0              1            0
2                      158         0              0            1
3                      155         0              1            0
4                      155         0              0            1
...                    ...       ...            ...          ...
99995                   90         0              1            0
99996                  100         0              1            0
99997                  155         0              0            1
99998                  100         0              1            0
99999                   90         0              1            0

       gender_Other  heart_disease_0  heart_disease_1
0                 0                0                1
1                 0                1                0
2                 0                1                0
3                 0                1                0
4                 0                0                1
...             ...              ...              ...
99995             0                1                0
99996             0                1                0
99997             0                1                0
99998             0                1                0
99999             0                1                0

[100000 rows x 12 columns]
```

Google Collaboratory Link: -

AIMLE1.ipynb

Conclusion: - Thus, we have successfully Collected, Cleaned, Integrated and Transformed our healthcare data.