Name: Md Mahinur Alam
Phone: +8801535133854
Email: mahinuralam213@gmail.com

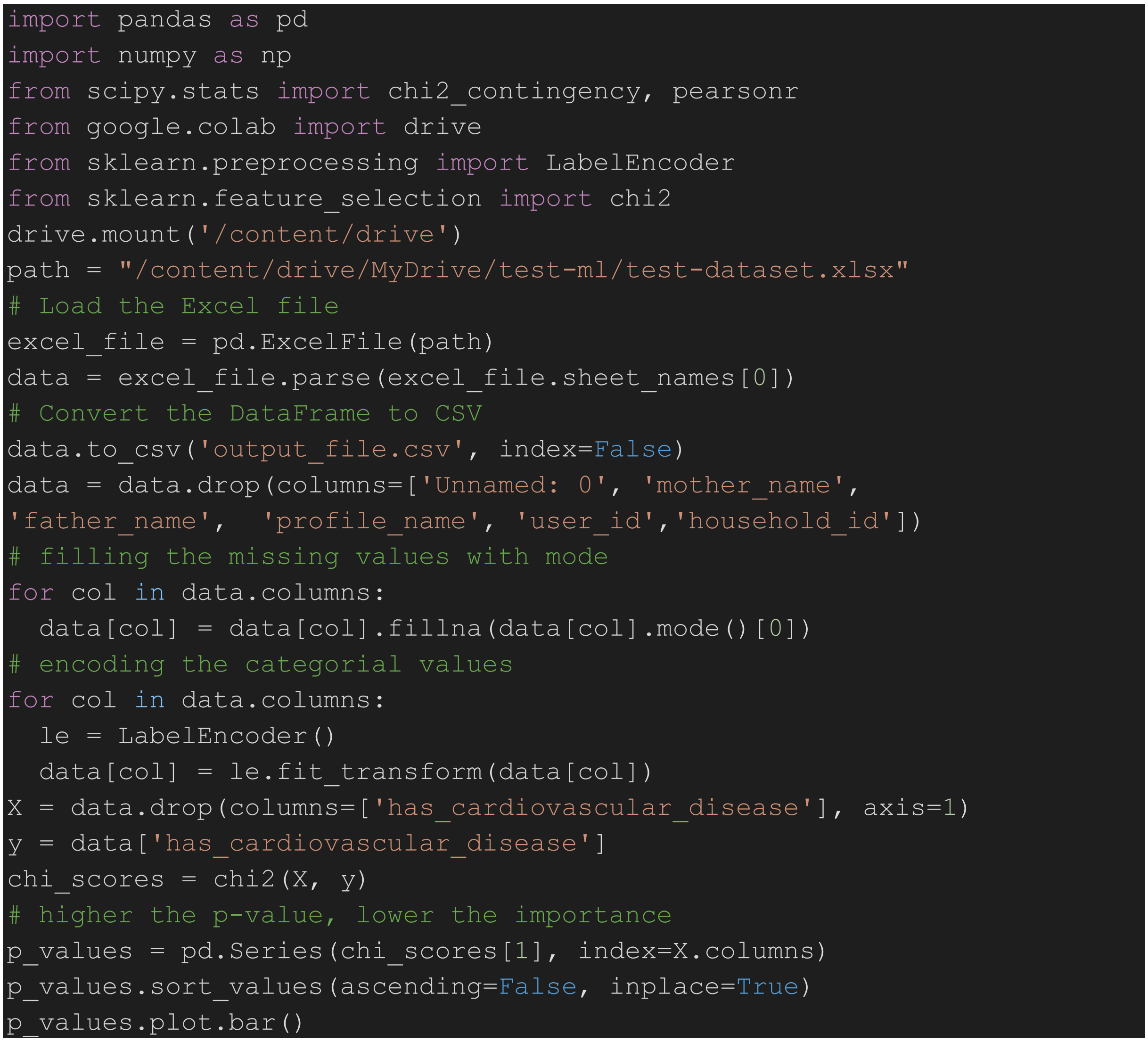
Null Hypothesis (H0): There is no significant association between the individual features and the presence of cardiovascular disease. In other words, each feature is independent of whether an individual has cardiovascular disease or not.

Alternative Hypothesis (H1): There is a significant association between at least one of the individual features and the presence of cardiovascular disease. In other words, at least one feature is not independent of whether an individual has cardiovascular disease or not.

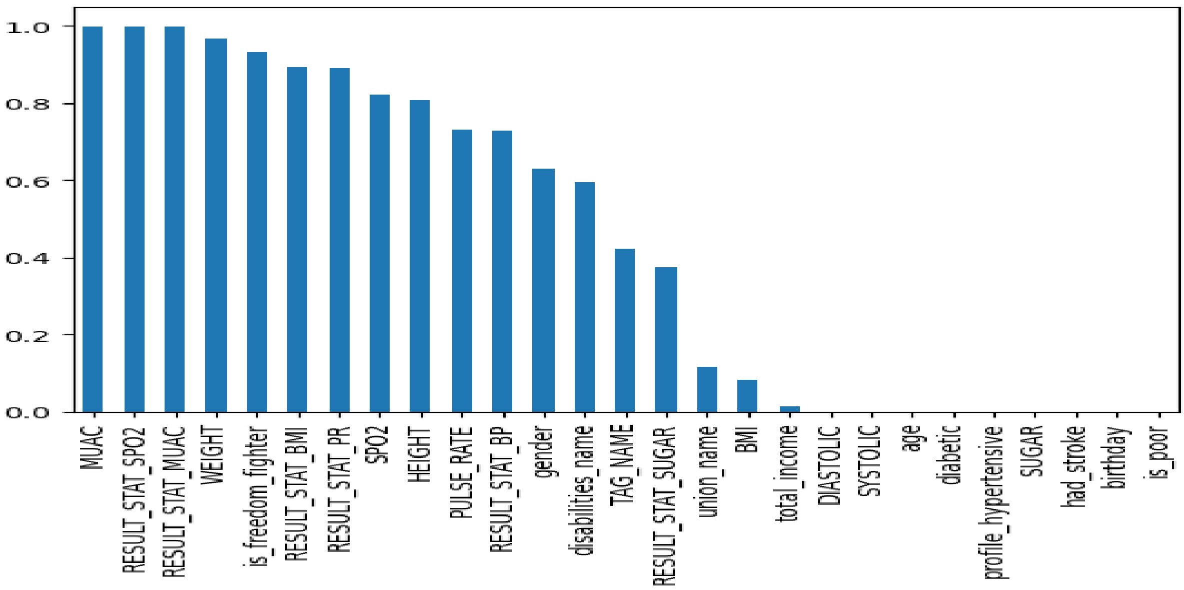
Apply Chi-Square: Chi-Square with a p-value threshold of 0.05 to achieve this goal. It is to ascertain the independence of the data. Given the dataset of two variables, with observed count OC and expected count EC. Chi-Square calculates how observed count (OC) and expected count (EC) differ.

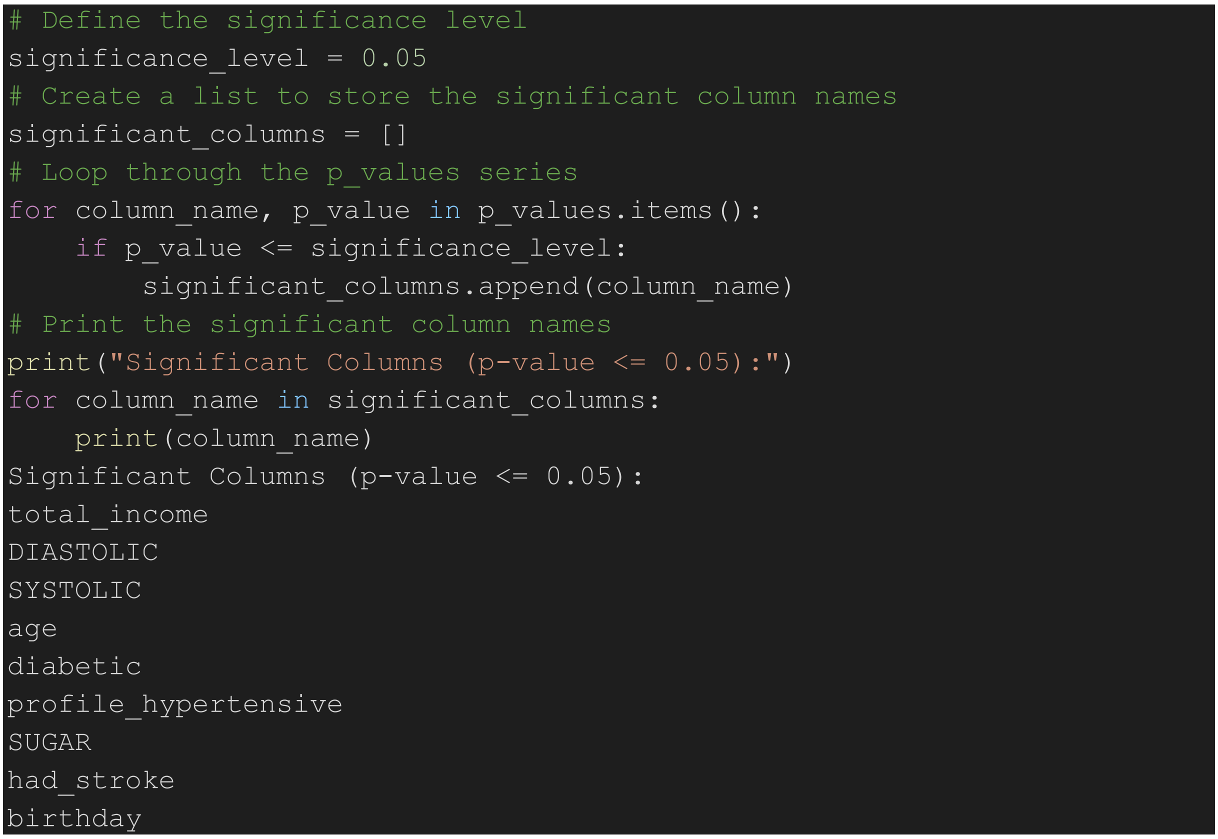
$$\chi_d^2 = \sum \frac{[OC_i - EC_i]^2}{EC_i},$$

d denotes the degree of freedom, EC is the expected value, and OC represents the observed value.



```
import pandas as pd
import numpy as np
from scipy.stats import chi2_contingency, pearsonr
from google.colab import drive
from sklearn.preprocessing import LabelEncoder
from sklearn.feature_selection import chi2
drive.mount('/content/drive')
path = "/content/drive/MyDrive/test-ml/test-dataset.xlsx"
# Load the Excel file
excel_file = pd.ExcelFile(path)
data = excel_file.parse(excel_file.sheet_names[0])
# Convert the DataFrame to CSV
data.to_csv('output_file.csv', index=False)
data = data.drop(columns=['Unnamed: 0', 'mother_name',
'father_name',  'profile_name', 'user_id','household_id'])
# filling the missing values with mode
for col in data.columns:
  data[col] = data[col].fillna(data[col].mode()[0])
# encoding the categorial values
for col in data.columns:
  le = LabelEncoder()
  data[col] = le.fit_transform(data[col])
X = data.drop(columns=['has_cardiovascular_disease'], axis=1)
y = data['has_cardiovascular_disease']
chi_scores = chi2(X, y)
# higher the p-value, lower the importance
p_values = pd.Series(chi_scores[1], index=X.columns)
p_values.sort_values(ascending=False, inplace=True)
p_values.plot.bar()
```





```
# Define the significance level
significance_level = 0.05
# Create a list to store the significant column names
significant_columns = []
# Loop through the p_values series
for column_name, p_value in p_values.items():
    if p_value <= significance_level:
        significant_columns.append(column_name)
# Print the significant column names
print("Significant Columns (p-value <= 0.05):")
for column_name in significant_columns:
    print(column_name)
Significant Columns (p-value <= 0.05):
total_income
DIASTOLIC
SYSTOLIC
age
diabetic
profile_hypertensive
SUGAR
had_stroke
birthday
```

With 95% confidence that is alpha = 0.05, we will check the calculated Chi-Square value falls in the acceptance or rejection region.

If the p-value is less than or equal to the chosen significance level (alpha) 0.05, we reject the null hypothesis (H0). This means we can believe that there is a statistically significant relationship or association between the variables being tested. So, the significant features are total_income, DIASTOLIC, SYSTOLIC, age, diabetic, profile_hypertensive, SUGA, had_stoke, and birthday in items of has_cardiovascular_disease is the target variable.