untitled10

September 5, 2024

[3]:
```python
import pandas as pd

# Load the dataset
data = pd.read_csv("C:/Users/sivasai/Downloads/archive (2)/
 ↪Heart_Disease_Prediction.csv")

# Display the first few rows of the dataset
data.head()
```

[3]:
```
   Age  Sex  Chest pain type   BP  Cholesterol  FBS over 120  EKG results  \
0   70    1                4  130          322             0            2
1   67    0                3  115          564             0            2
2   57    1                2  124          261             0            0
3   64    1                4  128          263             0            0
4   74    0                2  120          269             0            2

   Max HR  Exercise angina  ST depression  Slope of ST  \
0     109                0            2.4            2
1     160                0            1.6            2
2     141                0            0.3            1
3     105                1            0.2            2
4     121                1            0.2            1

   Number of vessels fluro  Thallium
0                        3         3
1                        0         7
2                        0         7
3                        1         7
4                        1         3
```

[4]:
```python
# Calculate mean, median, mode, standard deviation, and variance
mean = data.mean()
median = data.median()
mode = data.mode().iloc[0]  # Selecting the first mode value if multiple modes␣
 ↪exist
std_dev = data.std()
variance = data.var()
```

```python
# Display the results
print("Mean:\n", mean)
print("\nMedian:\n", median)
print("\nMode:\n", mode)
print("\nStandard Deviation:\n", std_dev)
print("\nVariance:\n", variance)
```

```
Mean:
 Age                        54.433333
Sex                         0.677778
Chest pain type             3.174074
BP                        131.344444
Cholesterol               249.659259
FBS over 120                0.148148
EKG results                 1.022222
Max HR                    149.677778
Exercise angina             0.329630
ST depression               1.050000
Slope of ST                 1.585185
Number of vessels fluro     0.670370
Thallium                    4.696296
dtype: float64

Median:
 Age                        55.0
Sex                         1.0
Chest pain type             3.0
BP                        130.0
Cholesterol               245.0
FBS over 120                0.0
EKG results                 2.0
Max HR                    153.5
Exercise angina             0.0
ST depression               0.8
Slope of ST                 2.0
Number of vessels fluro     0.0
Thallium                    3.0
dtype: float64

Mode:
 Age                        54.0
Sex                         1.0
Chest pain type             4.0
BP                        120.0
Cholesterol               234.0
FBS over 120                0.0
```

```
EKG results                  2.0
Max HR                     162.0
Exercise angina              0.0
ST depression                0.0
Slope of ST                  1.0
Number of vessels fluro      0.0
Thallium                     3.0
Name: 0, dtype: float64

Standard Deviation:
 Age                          9.109067
Sex                         0.468195
Chest pain type             0.950090
BP                         17.861608
Cholesterol                51.686237
FBS over 120                0.355906
EKG results                 0.997891
Max HR                     23.165717
Exercise angina             0.470952
ST depression               1.145210
Slope of ST                 0.614390
Number of vessels fluro     0.943896
Thallium                    1.940659
dtype: float64

Variance:
 Age                            82.975093
Sex                           0.219207
Chest pain type               0.902671
BP                          319.037051
Cholesterol                2671.467107
FBS over 120                  0.126669
EKG results                   0.995787
Max HR                      536.650434
Exercise angina               0.221795
ST depression                 1.311506
Slope of ST                   0.377475
Number of vessels fluro       0.890940
Thallium                      3.766157
dtype: float64
```

[6]:
```python
from scipy import stats

# Select the 'Cholesterol' column
cholesterol = data['Cholesterol'].dropna()

# Perform a one-sample t-test against 200
```

```
t_stat, p_value = stats.ttest_1samp(cholesterol, 200)

# Display the results
print(f"T-statistic: {t_stat}, P-value: {p_value}")

# Interpretation
alpha = 0.05  # Significance level

if p_value < alpha:
    print("Reject the null hypothesis: The average cholesterol is significantly␣
 ↪different from 200.")
else:
    print("Fail to reject the null hypothesis: There is no significant␣
 ↪difference from 200.")
```

```
T-statistic: 15.78727607352564, P-value: 3.4785009178906206e-40
Reject the null hypothesis: The average cholesterol is significantly different
from 200.
```

[8]:
```
import numpy as np

# Select the 'BP' column (Blood Pressure)
bp = data['BP'].dropna()

# Calculate the mean and standard error of the mean
mean_bp = np.mean(bp)
sem_bp = stats.sem(bp)

# Compute the 95% confidence interval
confidence_interval = stats.t.interval(0.95, len(bp)-1, loc=mean_bp,␣
 ↪scale=sem_bp)

# Display the results
print(f"Mean BP: {mean_bp}")
print(f"95% Confidence Interval for BP: {confidence_interval}")
```

```
Mean BP: 131.34444444444443
95% Confidence Interval for BP: (np.float64(129.2042899436035),
np.float64(133.48459894528537))
```

[10]:
```
import pandas as pd
import statsmodels.api as sm
import matplotlib.pyplot as plt

# Load the dataset
data = pd.read_csv("C:/Users/sivasai/Downloads/archive (2)/
 ↪Heart_Disease_Prediction.csv")
```

```python
# Display the first few rows of the dataset
data.head()
# Select the relevant features: Age (independent variable) and BP (dependent␣
↪variable)
X = data['Age']
y = data['BP']

# Add a constant to the independent variable (for the intercept)
X = sm.add_constant(X)

# Perform the linear regression
model = sm.OLS(y, X).fit()

# Display the model summary
print(model.summary())
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                     BP   R-squared:                       0.075
Model:                            OLS   Adj. R-squared:                  0.071
Method:                 Least Squares   F-statistic:                     21.59
Date:                Thu, 05 Sep 2024   Prob (F-statistic):           5.30e-06
Time:                        21:47:54   Log-Likelihood:                -1150.5
No. Observations:                 270   AIC:                             2305.
Df Residuals:                     268   BIC:                             2312.
Df Model:                           1
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const        102.1998      6.359     16.071      0.000      89.680     114.720
Age            0.5354      0.115      4.647      0.000       0.309       0.762
==============================================================================
Omnibus:                       21.308   Durbin-Watson:                   1.973
Prob(Omnibus):                  0.000   Jarque-Bera (JB):               26.516
Skew:                           0.599   Prob(JB):                     1.75e-06
Kurtosis:                       3.961   Cond. No.                         335.
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly
specified.
```

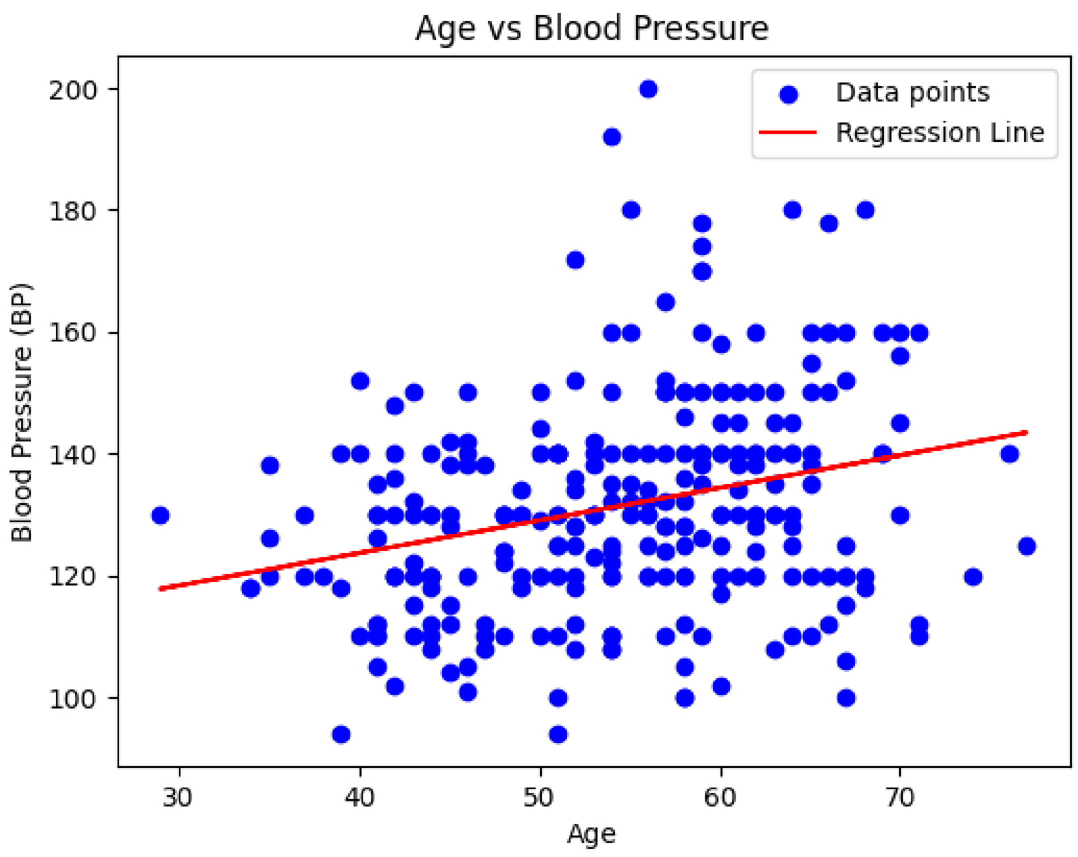
[11]:
```python
# Plotting the data points
plt.scatter(data['Age'], data['BP'], color='blue', label='Data points')
```

```
# Calculate the regression line
predictions = model.predict(X)

# Plotting the regression line
plt.plot(data['Age'], predictions, color='red', label='Regression Line')

# Add labels and title
plt.xlabel('Age')
plt.ylabel('Blood Pressure (BP)')
plt.title('Age vs Blood Pressure')

# Show legend and plot
plt.legend()
plt.show()
```



```
[ ]:
```