Notebook

April 23, 2024

Author: Alan H. Sadeeq Date: April 23, 2024

1 The Calculus of Best Fit: Application in Finding the Best Fitting Line

1.0.1 Using Optimization Techniques to Find the Best Fitting Line For a Simple Regression Model

1.0.2 Optimization in Calculus:

- Optimization in calculus deals with finding the maximum or minimum of a function. This could involve finding the maximum profit, minimum cost, optimal shape, etc., by analyzing the behavior of a function.
- Methods like finding critical points (where the derivative is zero), using the first and second derivative tests, or applying optimization algorithms such as gradient descent are employed.

1.0.3 Least Squares Regression Line in Statistics:

- The least squares regression line is a statistical technique used to model the relationship between two variables by fitting a linear equation to observed data.
- It minimizes the sum of the squared differences between the observed and predicted values (residuals) of the dependent variable (usually denoted as y) for given values of the independent variable (usually denoted as x).

2 Least Squares Regression Model

[6]:
```
# importing the necessary libraries

import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
from sklearn.linear_model import LinearRegression
```

2.1 We would like to create a model depicting the relationship between blood pressure and an individual's age.

We want to find the best fitting line, then we can make predictions and ask questions such as, "What's the change in blood pressure for each one year increase in age?"

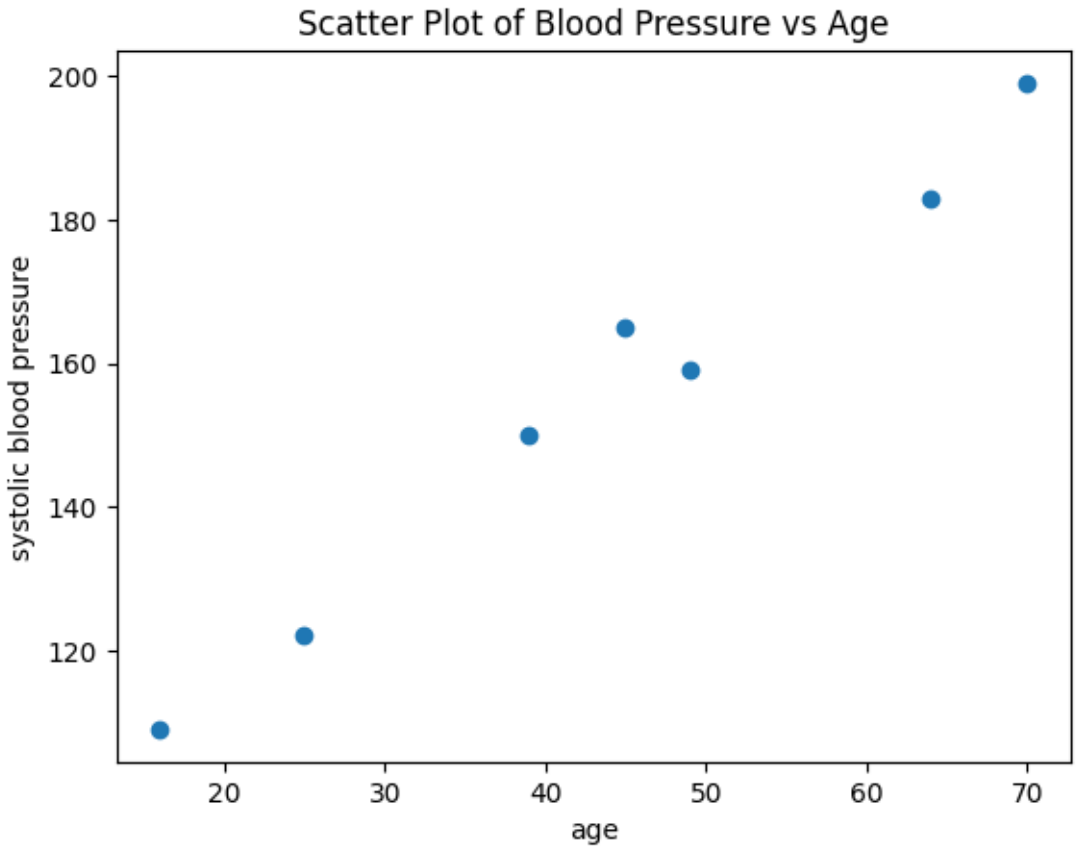
The answer is going to be a mathematics model for this blood pressure problem and we going to use least square regression line to find this model.

Given a set of data we can model it with a stright line, with a parabola, with a cubic function, with a trigonometric functions, with an exponential functions. All kind of models exist it depends on the data.

In our problem we are going to use the linear least square model.

[14]:
```
x = np.array([16, 25, 39, 45, 49, 64, 70]).reshape(-1, 1)  # Reshape x into a␣
↪column vector
y = np.array([109, 122, 150, 165, 159, 183, 199])

z = plt.scatter(x, y)
plt.xlabel('age')
plt.ylabel('systolic blood pressure')
plt.title('Scatter Plot of Blood Pressure vs Age')
plt.show()
```



2.1.1 We want to find the best fitting line. The function of a straight line in the slope intercept form is:

$$f(x) = ax + b$$

2.1.2 Now we want to find the parameters a and b.

$$RSS(a,b) = \sum_{i=1}^{n}(f(x_i) - \hat{y}_i)^2 = \sum_{i=1}^{n}(ax_i + b - \hat{y}_i)^2$$

2.1.3 RSS is an example of cost function also called loss function or error function, the idea here is to minimize RSS so it produces the least square regression line.

2.1.4 We have formulas for a and b that produces the least RSS:

$$a = \frac{n\sum_{i=1}^{n} x_i y_i - (\sum_{i=1}^{n} x_i)(\sum_{i=1}^{n} y_i)}{n\sum_{i=1}^{n} x_i^2 - (\sum_{i=1}^{n} x_i)^2}$$

$$b = \frac{\sum_{i=1}^{n} y_i - a\sum_{i=1}^{n} x_i}{n}$$

2.1.5 We can use a table to find the values of a and b but we want to answer a deeper question! Where does those formulas come from?

2.1.6 How Optimization Helps in Least Squares Regression:

- Optimization techniques are used to find the coefficients (slope and intercept) of the regression line that minimize the sum of squared residuals.
- By minimizing this sum, the regression line is fitted as closely as possible to the observed data points, providing the best linear approximation of the relationship between the variables.
- The process involves taking derivatives with respect to the coefficients and setting them equal to zero to find the optimal values that minimize the objective function (the sum of squared residuals). This often involves calculus techniques such as solving systems of linear equations or using matrix algebra.
- In cases where analytical solutions are not feasible, numerical optimization algorithms like gradient descent can be employed to iteratively find the optimal solution.

2.1.7 RSS is a function of two variables a and b. We going to take the partial derivates of RSS with respect to a and b respectively and set them equal to 0 to minimize the values of a and b.

$$SSR_a(a,b) = 2a\sum_{i=1}^{n} x_i^2 + 2b\sum_{i=1}^{n} x_i - 2\sum_{i=1}^{n} x_i y_i = 0$$

$$SSR_b(a,b) = 2a\sum_{i=1}^{n} x_i + 2nb - 2\sum_{i=1}^{n} y_i = 0$$

2.1.8 Now we arrived at the formulas for a and b.

2.1.9 We can use the second partial test to varify that indeed that yelds the minimum.

2.1.10 Now back to our original question, "What's the change in blood pressure for each one year increase in age?" in other word "How much blood pressure inscreases each year on average?"

We are going to use pre-build functions to calculate the best fitting line instead of using tables.

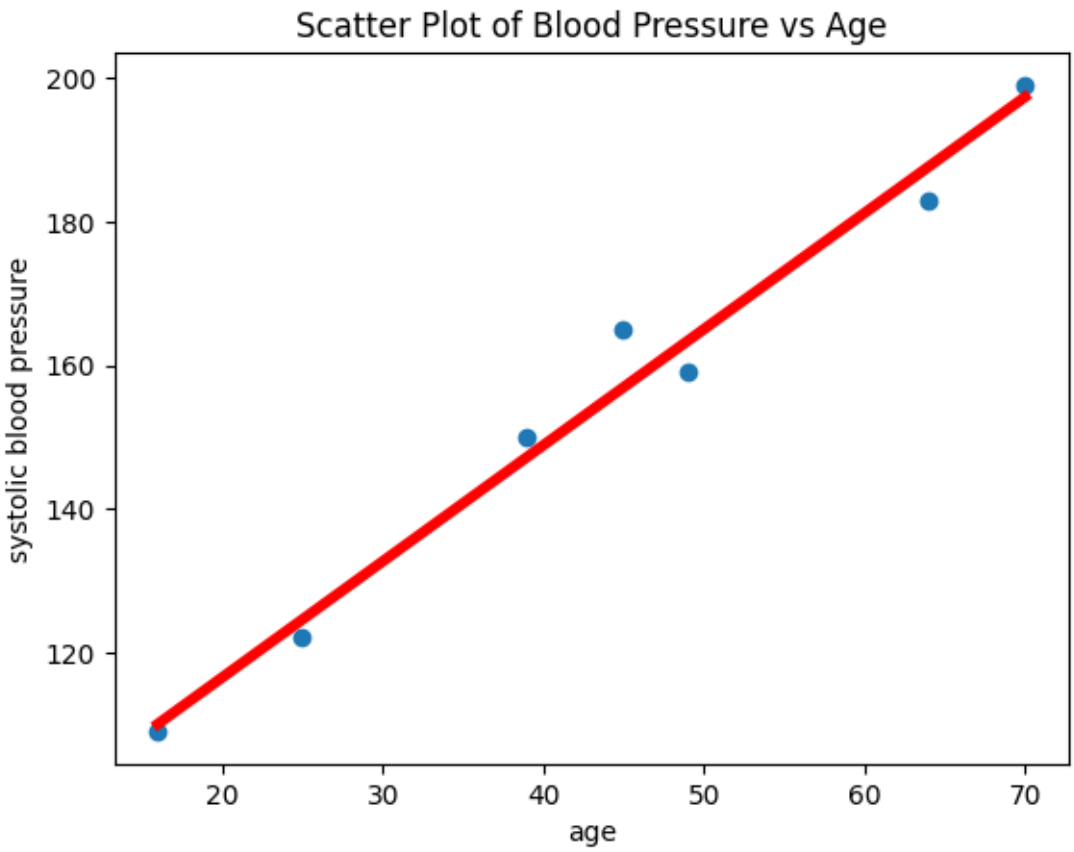
[15]:
```
regression = LinearRegression()
regression.fit(x, y)

# Get the coefficients (slope and intercept)
slope = regression.coef_[0]
intercept = regression.intercept_

print("Slope (a):", slope)
print("Intercept (b):", intercept)
```

```
Slope (a): 1.6170774647887323
Intercept (b): 84.13430583501005
```

[18]:
```
z = plt.scatter(x, y)
plt.plot(x, regression.predict(x), color='red', linewidth=4)
plt.xlabel('age')
plt.ylabel('systolic blood pressure')
plt.title('Scatter Plot of Blood Pressure vs Age')
plt.show()
```



2.2 Conclusion

2.2.1 For each one year increase in age (x), the blood pressure increases by approximately 1.6