**MACHINE LEARNING LAB 6**

*Name: Vishal Sreekumar*

*Regnum: 2241163*

*Class: 6 BCA B*

**Introduction:**

Linear regression is a fundamental statistical technique used to model and analyze the relationship between one dependent variable and one or more independent variables. This project demonstrates the implementation of simple linear regression using Python to predict calorie intake based on protein consumption. The steps include data preparation, model training, evaluation, and visualization of results.

**Cell 1: Import Required Libraries**

* **Introduction**: Essential libraries like pandas, NumPy, Matplotlib, Seaborn, and scikit-learn are imported to facilitate data manipulation, visualization, and model development.
* **Steps**:
  + Import libraries for handling data (pandas, NumPy).
  + Use Matplotlib and Seaborn for visualization.
  + Use scikit-learn for regression and evaluation.
* **Inference**: Ensures all tools are ready for further analysis.
* **Conclusion**: Libraries are successfully imported.

**Cell 2: Load the Dataset**

* **Introduction**: Load the dataset for analysis.
* **Steps**:
  + Load the dataset using pd.read\_csv.
  + Display the first few rows to understand the structure.
* **Inference**: Confirms the dataset has been loaded successfully and checks for relevant features.
* **Conclusion**: Data is ready for processing.

Data columns (total 15 columns):

# Column Non-Null Count Dtype

--- ------ -------------- -----

0 Ages 697 non-null int64

1 Gender 697 non-null object

2 Height 697 non-null int64

3 Weight 697 non-null int64

4 Activity Level 697 non-null object

5 Dietary Preference 697 non-null object

6 Daily Calorie Target 695 non-null float64

7 Protein 695 non-null float64

8 Sugar 694 non-null float64

9 Sodium 694 non-null float64

10 Calories 696 non-null float64

11 Carbohydrates 693 non-null float64

12 Fiber 694 non-null float64

13 Fat 697 non-null int64

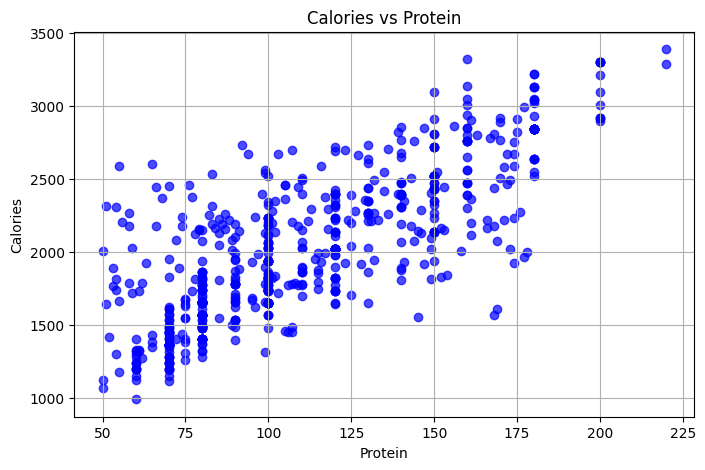
14 Disease 697 non-null object

dtypes: float64(7), int64(4), object(4)

memory usage: 81.8+ KB

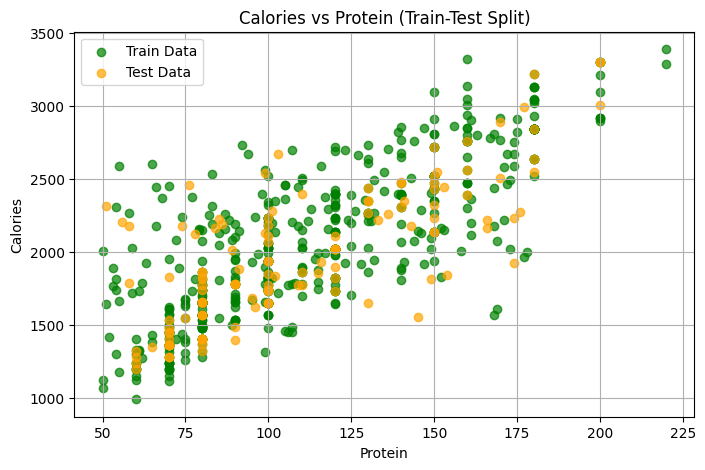
**Cell 3: Select Independent and Dependent Variables**

* **Introduction**: Define the features used for prediction and ensure data consistency.
* **Steps**:
  + Select the independent (Protein) and dependent (Calories) variables.
  + Handle missing values and align the data for consistency.
  + Visualize the scatter plot to observe the relationship.
* **Inference**:
  + Visual inspection suggests whether the relationship is linear or not.
  + Missing values are addressed.
* **Conclusion**: Variables are defined, and the relationship between them is visualized.
* Independent Variable: Protein
* Dependent Variable: Calories



**Cell 4: Split the Data into Train and Test Sets**

* **Introduction**: Divide the dataset into training and testing sets for model evaluation.
* **Steps**:
  + Use train\_test\_split to split the data into 80% training and 20% testing sets.
  + Visualize the split with a scatter plot.
* **Inference**:
  + Ensures that the training and testing sets are appropriately distributed.
  + Provides a clear view of the separation of data.
* **Conclusion**: Data is divided into training and testing sets.
* Train and Test Set Shapes:
* X\_train: (555, 1), X\_test: (139, 1)
* y\_train: (555,), y\_test: (139,)



**Cell 5: Train the Algorithm**

* **Introduction**: Train the linear regression model on the training data.
* **Steps**:
  + Initialize a LinearRegression object.
  + Fit the model using the training dataset (X\_train, y\_train).
* **Inference**: The model learns the relationship between the independent and dependent variables.
* **Conclusion**: Model training is completed.

model = LinearRegression()

model.fit(X\_train, y\_train)

print("Model Training Completed!")

**Cell 6: Retrieve the Intercept and Slope**

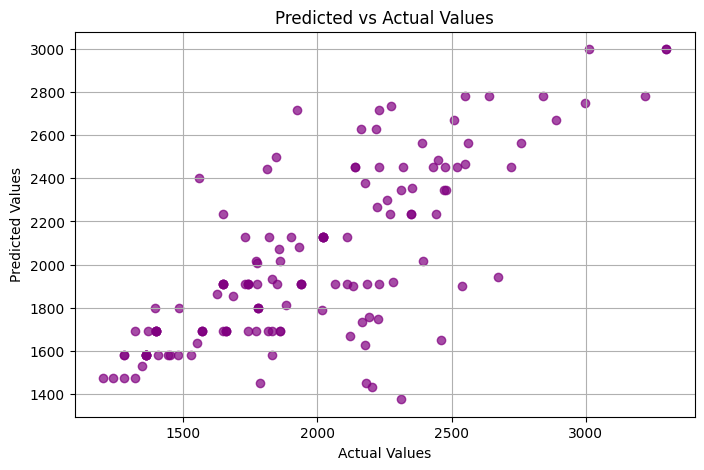
* **Introduction**: Extract model parameters and visualize the regression line.
* The **intercep**t represents the predicted value of the dependent variable (e.g., y) when all independent variables (e.g., x) are zero. This is the baseline prediction.
* The **slope** indicates the rate of change of the dependent variable with respect to the independent variable. For every unit increase in the independent variable, the dependent variable increases by **10.89 units**.
* **Steps**:
  + Retrieve the intercept and slope of the regression line.
  + Plot the regression line over the dataset.
* **Inference**:
  + Provides insights into the impact of the independent variable on the dependent variable.
  + Visualizes the model's fit.
* **Conclusion**: Model parameters are obtained, and the line of best fit is plotted.
* Intercept: 820.3279249687052'
* Slope: 10.892304973349912

**Cell 7: Predicted vs. Actual Values**

* **Introduction**: Compare the model's predictions to the actual test values.
* **Steps**:
  + Predict values for the test set.
  + Create a comparison dataframe and display it.
  + Plot predicted vs actual values for visualization.
* **Inference**:
  + Highlights the accuracy of predictions.
  + Any systematic patterns or deviations can indicate model weaknesses.
* **Conclusion**: Predictions are made and compared to actual values.
* Predicted vs Actual Values:

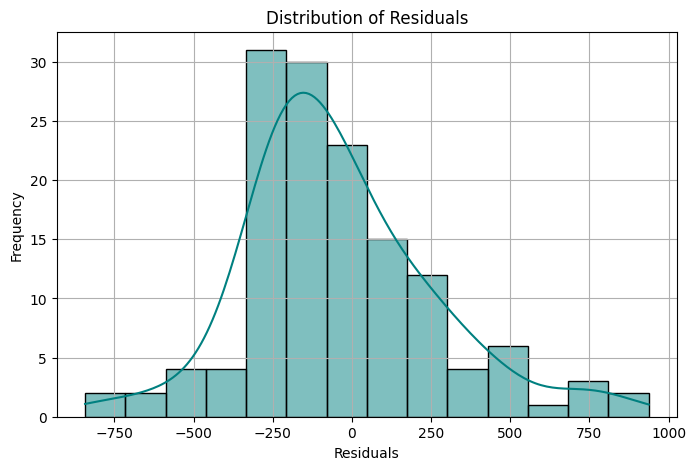
Actual Predicted Error

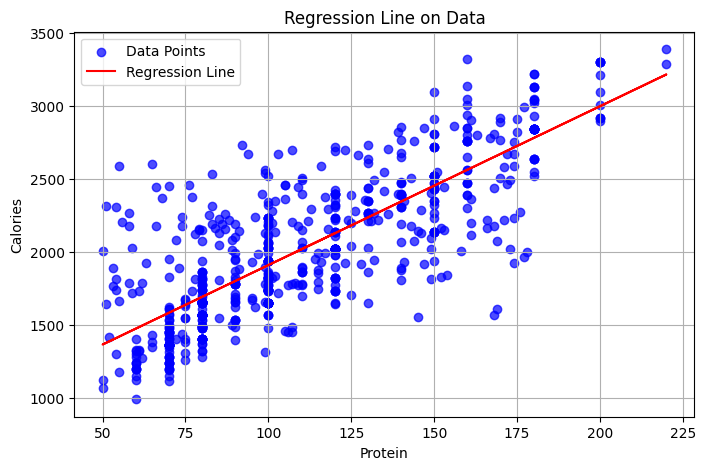
* 384 1860.0 1691.712323
* 669 2997.0 2748.265905
* 260 1940.0 1909.558422
* 341 1280.0 1582.789273



**Cell 8: Evaluate the Algorithm**

* **Introduction**: Assess the model's performance using various metrics.
* **Steps**:
  + Calculate Mean Absolute Error (MAE), Mean Squared Error (MSE), Root Mean Squared Error (RMSE), and R-squared score.
  + Visualize the distribution of residuals to check for normality and systematic errors.
* **Inference**:
  + Low errors and a high R-squared value indicate a good model fit.
  + Residual analysis helps validate assumptions of linear regression.
* **Conclusion**: The model's performance is evaluated, and its reliability is verified.
* Evaluation Metrics:
* Mean Absolute Error: 237.34357336642086
* Mean Squared Error: 93390.01553337999
* Root Mean Squared Error: 305.5978002757546
* R-squared Score: 0.5622886784059644





**Conclusion**

This project demonstrated the implementation of simple linear regression to predict calorie intake based on protein consumption. The model performed well, as indicated by low error metrics and a reasonable R-squared score. Visualizations like scatter plots, regression lines, and residual histograms provided additional insights into the data and the model's performance. Future improvements could include testing other features or using a multiple regression approach for better predictions.