HyperionDev

Machine Learning and Simple Linear Regression

September 2024

Data Science Session Housekeeping

- The use of disrespectful language is prohibited in the questions, this is a supportive, learning environment for all - please engage accordingly.
- No question is daft or silly - **ask them!**
- There are **Q&A sessions** midway and at the end of the session, should you wish to ask any follow-up questions. Moderators are going to be answering questions as the session progresses as well.
- If you have any questions outside of this lecture, or that are not answered during this lecture, please do submit these for upcoming Academic Sessions. You can submit these questions here: **Questions**

Data Science Session Housekeeping cont.

- For all **non-academic questions**, please submit a query: **www.hyperiondev.com/support**
- Report a **safeguarding** incident: **www.hyperiondev.com/safeguardreporting**
- We would love your **feedback** on lectures: **Feedback on Lectures**

Learning Outcomes

Understand and implement simple linear regression models using Python and scikit-learn.

- Define simple linear regression and its purpose
- Interpret the mathematical equation and assumptions of simple linear regression
- Implement and evaluate simple linear regression models using Python

HyperionDev

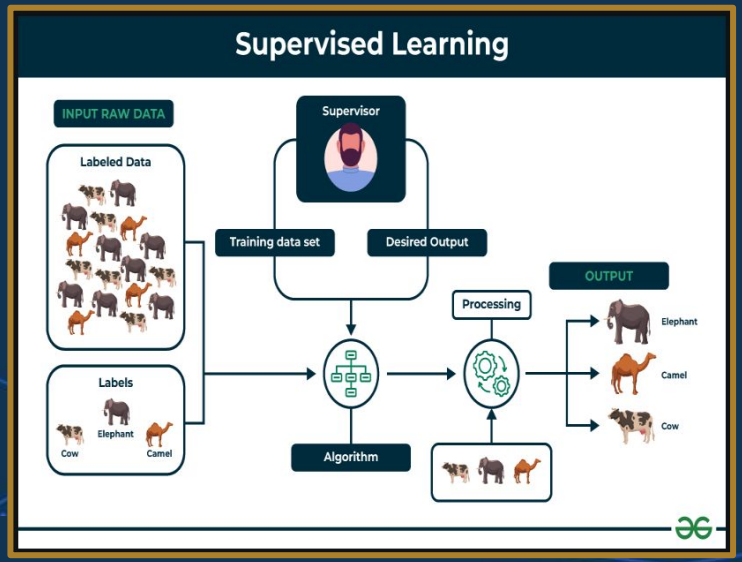
Introduction to Machine Learning

- Machine learning is a way of teaching computers to learn and improve from experience without being explicitly programmed.
- It allows computers to automatically learn and adapt based on data.

HyperionDev

Types of machine learning

- **Supervised learning:** The computer learns from labelled data, where both input and output data are provided.

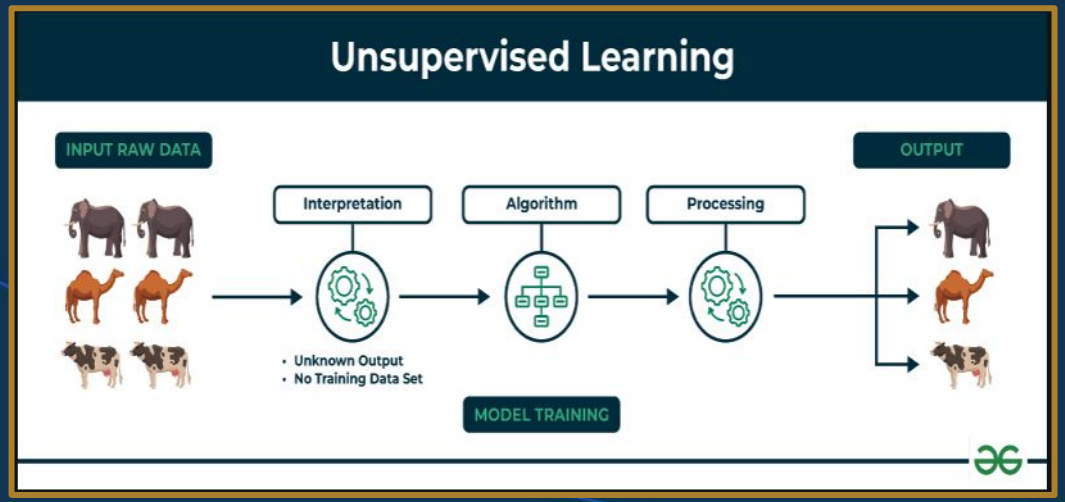


Source: [geeksforgeeks]

HyperionDev

Types of machine learning

- **Unsupervised learning:** The computer learns from unlabeled data, discovering hidden patterns or structures on its own.

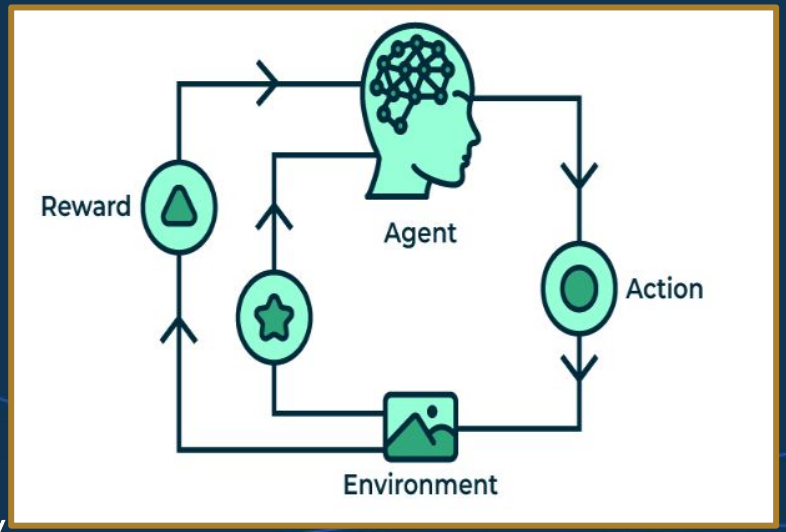


Source: [geeksforgeeks]

HyperionDev

Types of machine learning

- **Reinforcement learning:** The computer learns through interaction with an environment, receiving rewards or penalties for its actions.



Source: geeksforgeeks

HyperionDev

Some Applications of machine learning

- **Spam email filtering:** Identifying and separating spam emails from regular emails.
- **Image recognition:** Recognizing objects, faces, or scenes in images.
- **Recommender systems:** Suggesting products, movies, or songs based on user preferences.

Supervised Learning

- In supervised learning, the algorithm learns from labelled data, which consists of input-output pairs.
- The goal is to learn a function that maps input data to the correct output labels.

HyperionDev

Types of Supervised Learning

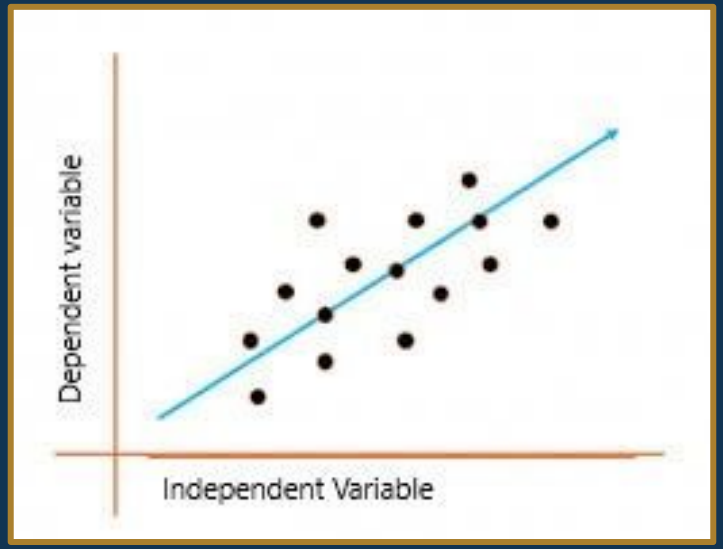
- **Regression:** Predicting continuous numerical values, such as house prices or stock prices.
- **Classification:** Predicting discrete categories or classes, such as whether an email is spam or not.

Supervised Learning Algorithms

- **Linear regression:** Fitting a straight line to data points to make predictions.
- **Logistic regression:** Predicting binary outcomes, such as yes/no or true/false.
- **Decision trees:** Making decisions based on a series of questions or conditions.
- **Support vector machines (SVM):** Finding the best boundary to separate different classes.
- **Neural networks:** Mimicking the structure and function of the human brain to learn complex patterns.

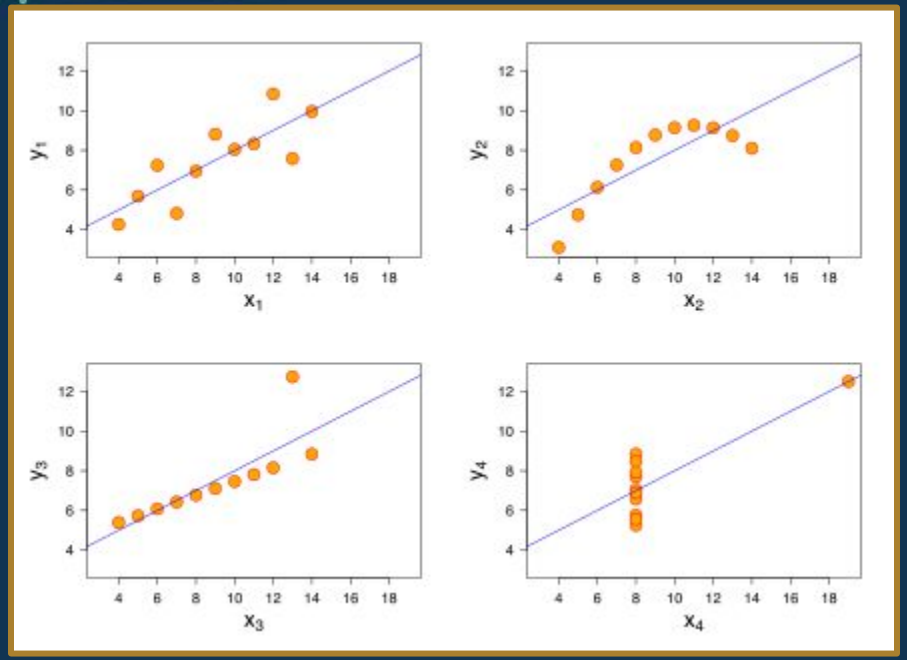
Simple Linear Regression

- Simple linear regression is a method to study the relationship between two variables: an independent variable (x) and a dependent variable (y).
- It helps us understand how changes in the independent variable affect the dependent variable.



Source: [Analytics Vidhya]

HyperionDev



Source: [Wikipedia]

HyperionDev

Purpose of Simple Linear Regression

- To find the best-fitting straight line that describes the relationship between x and y.
- This line can be used to make predictions about the dependent variable based on new values of the independent variable.

HyperionDev

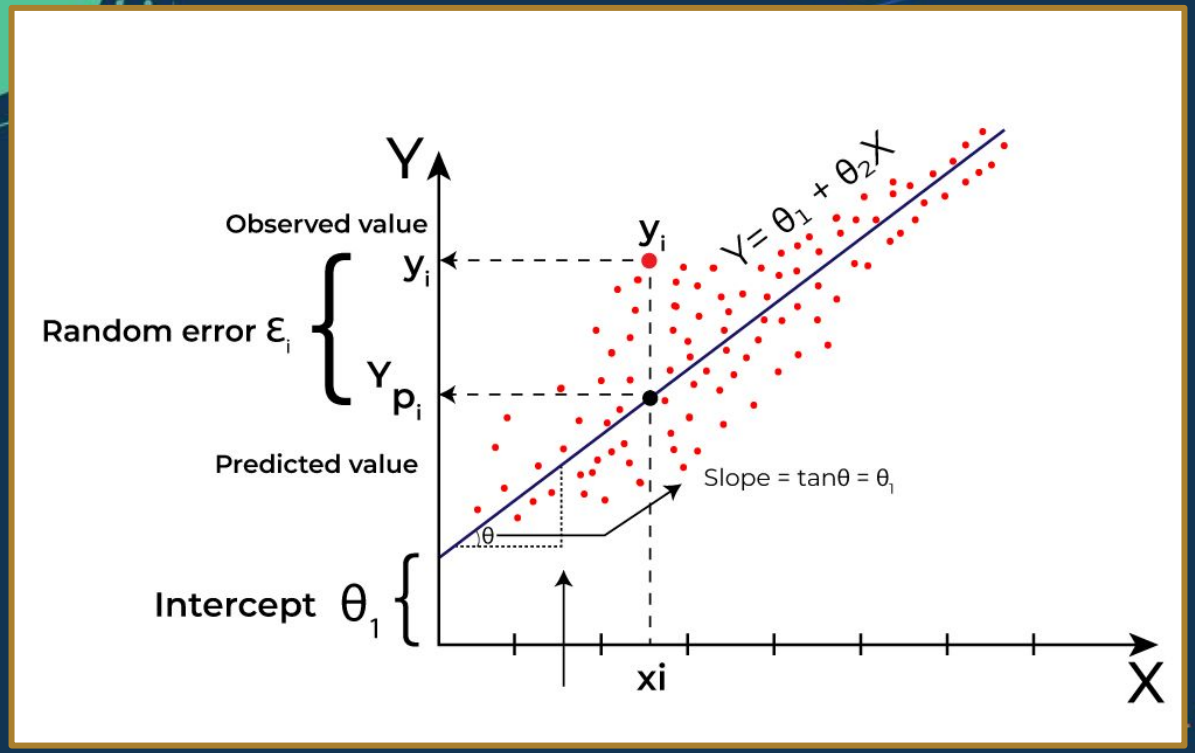
Applications of Simple Linear Regression

- **Sales forecasting:** Predicting future sales based on historical data.
- **Price prediction:** Estimating the price of a product based on its features.
- **Trend analysis:** Identifying trends or patterns in data over time.

HyperionDev

Math behind Simple Linear Regression

- The equation is written as: $y = \beta_0 + \beta_1 x + \varepsilon$
  - β_0 is the intercept, representing the value of y when x is zero.
  - β_1 is the slope, indicating how much y changes for a one-unit increase in x.
  - ε is the error term, accounting for the variability in y that cannot be explained by x.

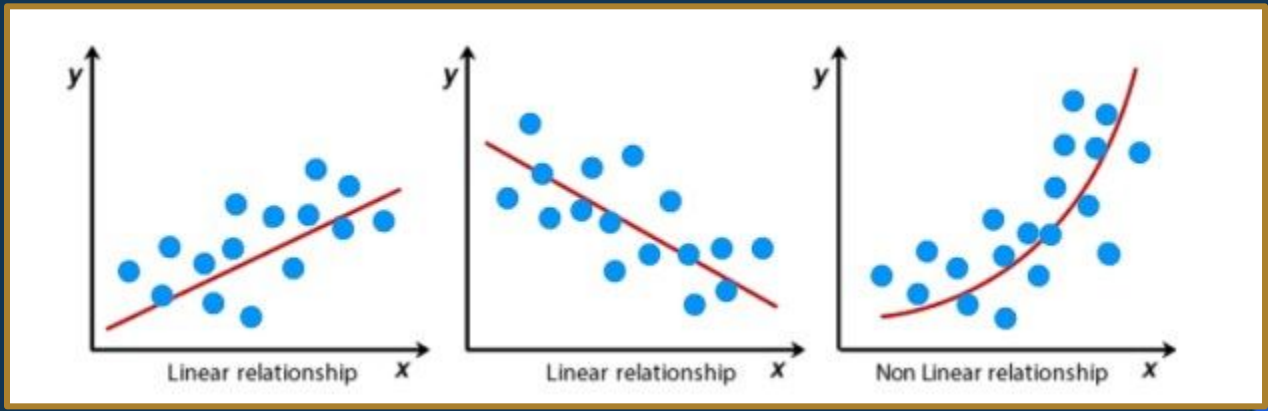


Source: [geeksforgeeks]

HyperionDev

Assumptions and Limitations of Simple Linear Regression

- **Linearity:** The relationship between x and y should be linear.

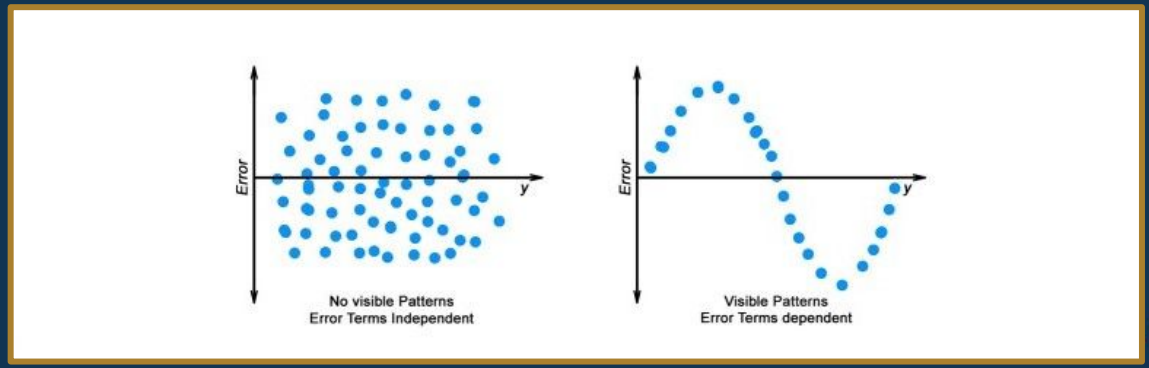


Source: Analytics Vidhya

HyperionDev

Assumptions and Limitations of Simple Linear Regression

- **Independence:** The observations should be independent of each other.

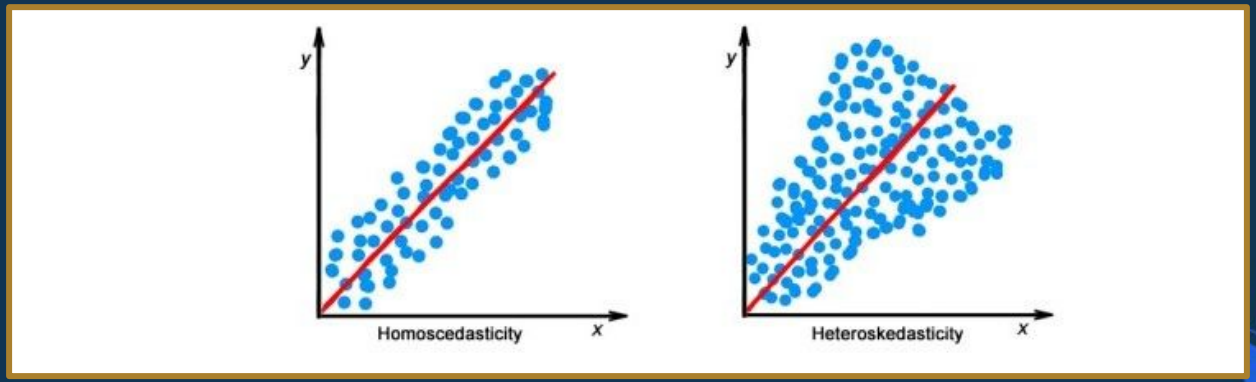


Source: Analytics Vidhya

HyperionDev

Assumptions and Limitations of Simple Linear Regression

- **Homoscedasticity:** The variability of y should be constant across all values of x.

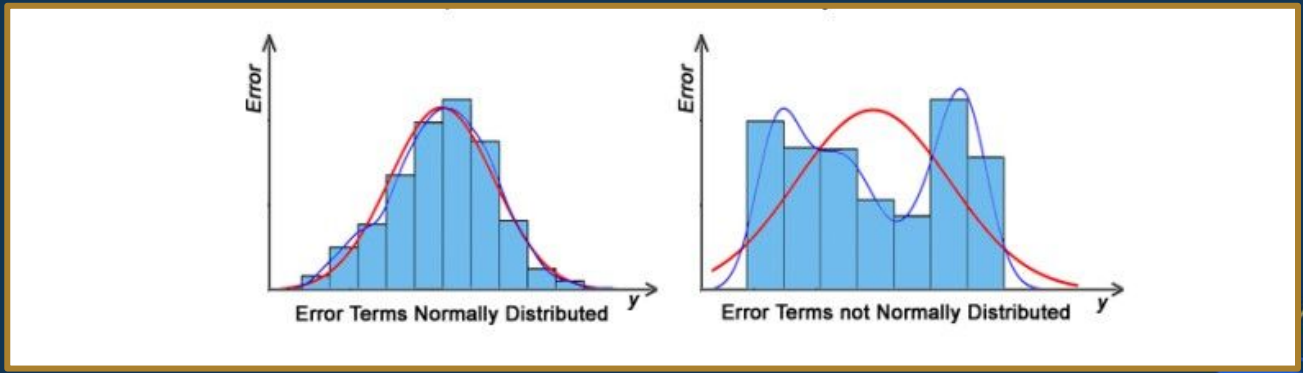


Source: Analytics Vidhya

HyperionDev

Assumptions and Limitations of Simple Linear Regression

- **Normality:** The errors should be normally distributed.



Source: Analytics Vidhya

HyperionDev

Implementing Simple Linear Regression

HyperionDev

Scikit-learn

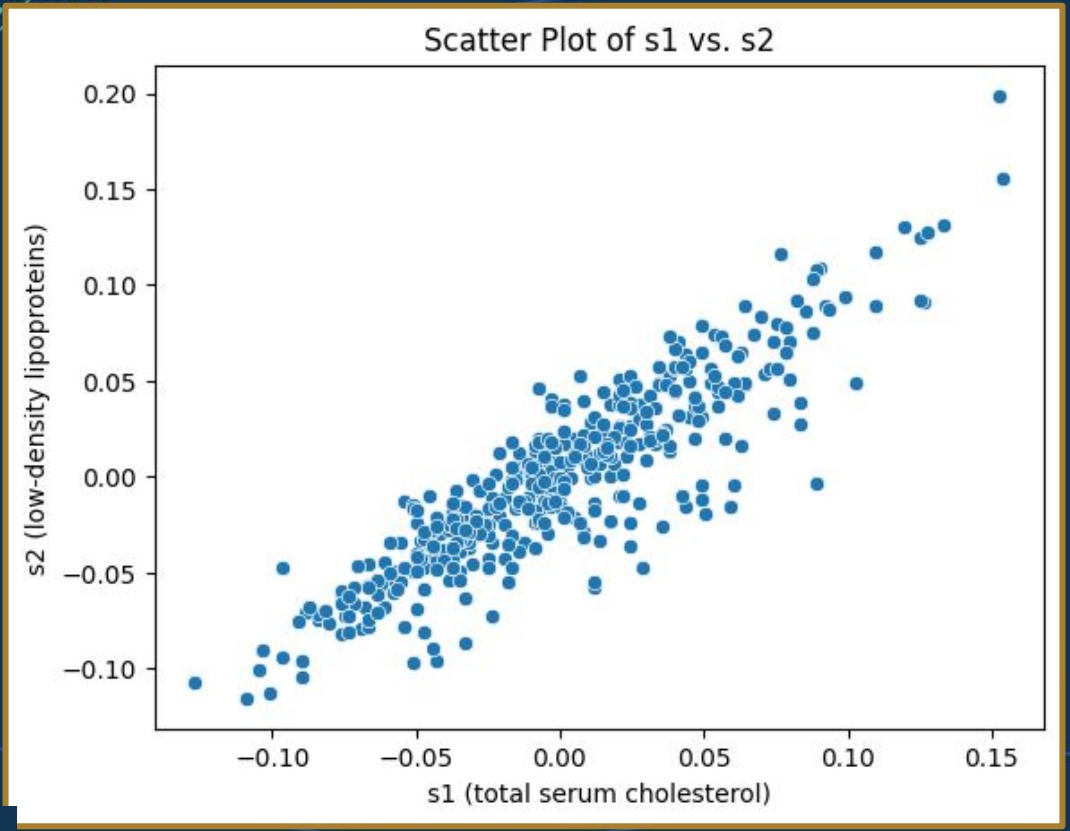
- Scikit-learn is a popular Python library for machine learning.
- It provides simple and efficient tools for data analysis and modelling.

```
from sklearn.datasets import load_diabetes
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error, r2_score
```

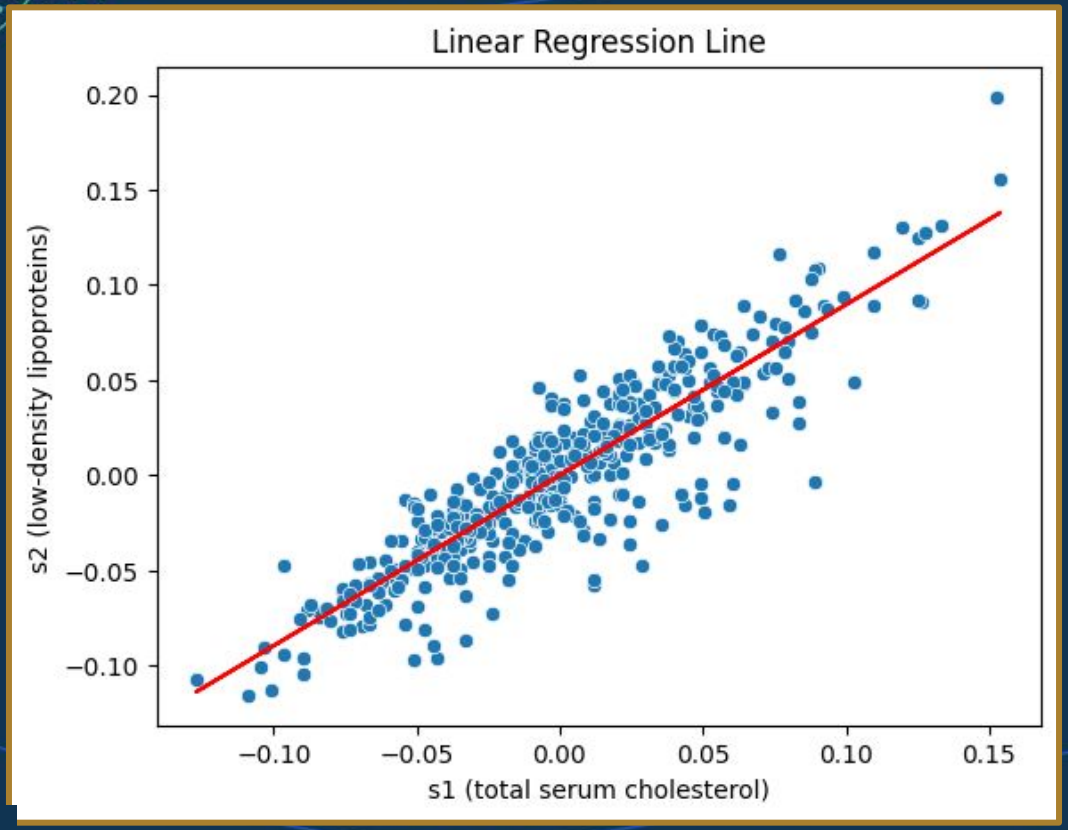
Loading the Diabetes Dataset

- We'll use the built-in diabetes dataset from scikit-learn for our example.
- The dataset contains information about various medical predictors and a quantitative measure of disease progression.

```
df = load_diabetes(as_frame=True).data
```



HyperionDev



HyperionDev

```
Evaluation Metrics:
Mean Squared Error (MSE): 0.00044342882373426217
R-squared (R2) Score: 0.8040044599094562
```

Interpretation of Results

- Scatter plot:
  - The scatter plot visualises the relationship between s1 (total serum cholesterol) and s2 (low-density lipoproteins).
  - It helps assess the linearity and spread of the data points.
- Linear regression line:
  - The red line represents the best-fit line obtained from the linear regression model.
  - It shows the predicted relationship between s1 and s2 based on the trained model.

Evaluation Metrics

- Mean Squared Error (MSE):
  - MSE measures the average squared difference between the predicted and actual values.
  - A lower MSE indicates better model performance.
- R-squared (R^2) score:
  - R^2 represents the proportion of variance in the target variable that can be explained by the model.
  - An R^2 value closer to 1 indicates a better fit of the model to the data.

Evaluation Metrics

- $R^2 = 1 - SS_{error}/SS_{total}$
- Where:
  - $SS_{error} = \Sigma(y_i - \hat{y}_i)^2$
  - $SS_{total} = \Sigma(y_i - E[y_i])^2$

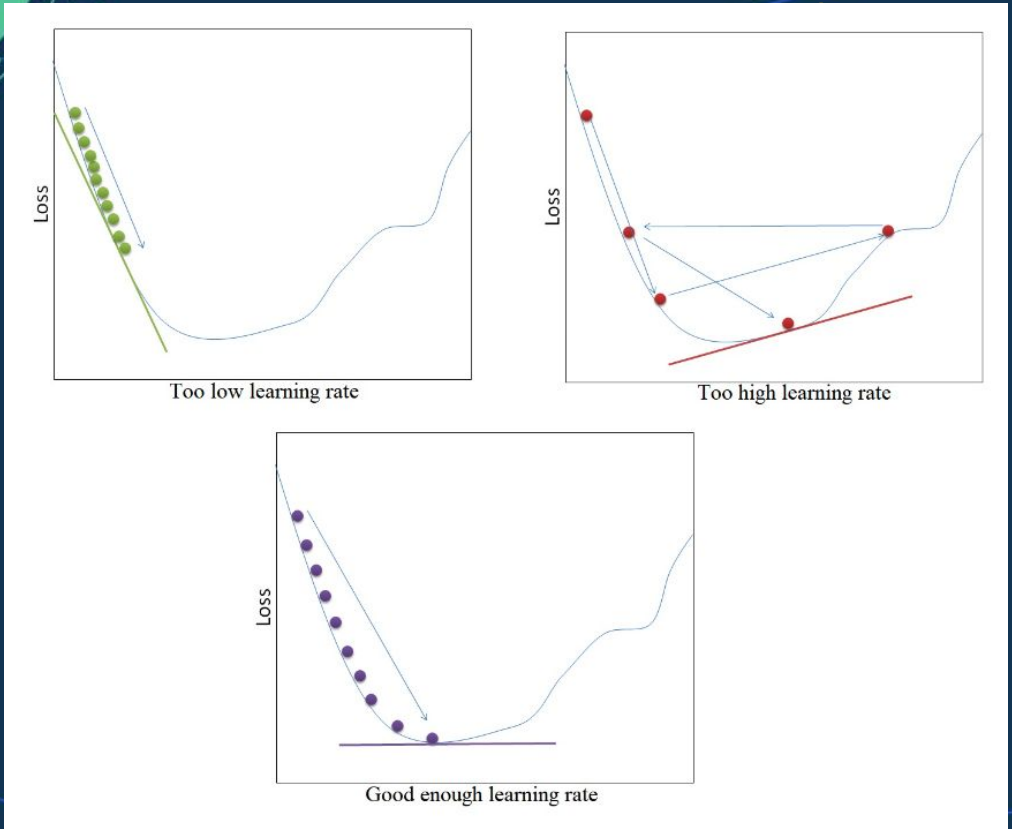
HyperionDev

Evaluation Metrics

- **Accuracy** is another commonly used metric for evaluating the performance of a machine learning model, particularly in classification problems.
  - **Accuracy = (Number of correct predictions) / (Total number of predictions) * 100%**
- While accuracy is more suitable for classification tasks, metrics like Mean Squared Error (MSE) and R-squared (R^2) are used for regression problems.

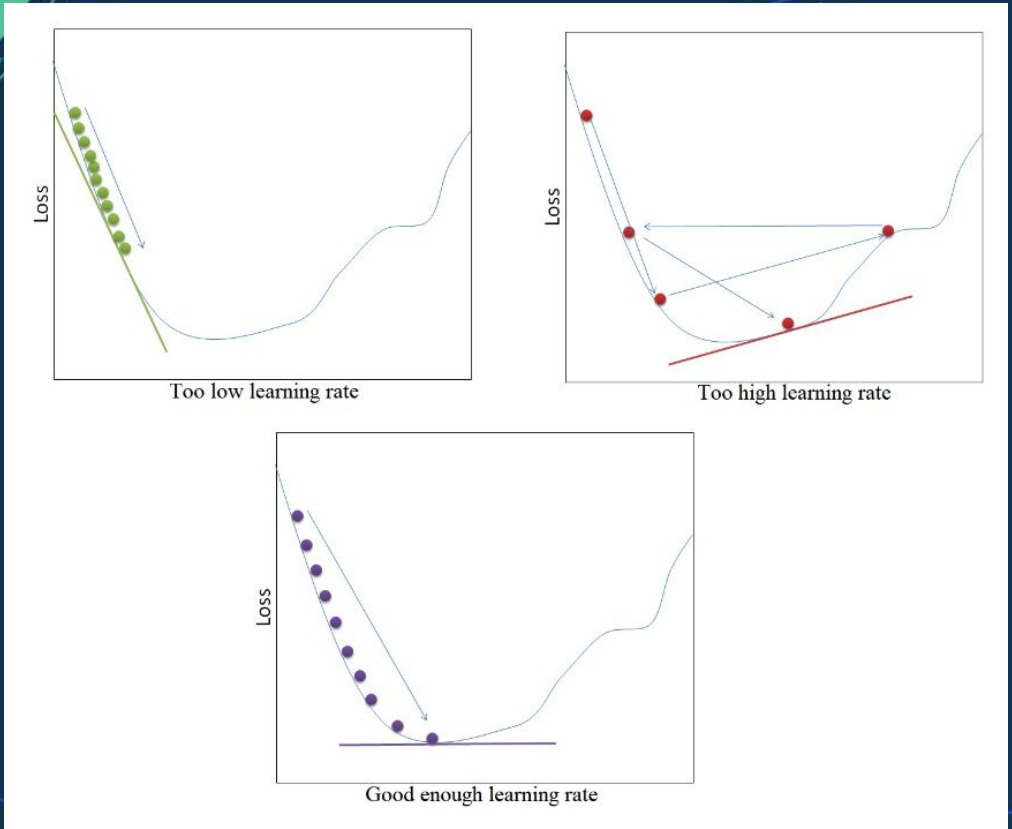
Parameter Tuning

- **Parameter tuning** is the process of finding the optimal values for a model's hyperparameters to improve its performance.
- **Hyperparameters are settings** that are not learned from the data but are set before training the model.
  - Examples of hyperparameters in linear regression include the **learning rate**, **regularization strength**, and the **number of iterations**.



Source: Analytics Vidhya

HyperionDev



Source: Analytics Vidhya

HyperionDev

Questions and Answers

HyperionDev

Thank you for attending

HyperionDev