



Regularization in Machine Learning

Last Updated : 05 Aug, 2024

While developing machine learning models you must have encountered a situation in which the training accuracy of the model is high but the validation accuracy or the testing accuracy is low. This is the case which is popularly known as overfitting in the domain of [machine learning](#). Also, this is the last thing a machine learning practitioner would like to have in his model.

In this article, we will explore a powerful technique known as Regularization in Python, which helps to mitigate the problem of overfitting. Regularization introduces a penalty for more complex models, effectively reducing their complexity and encouraging the model to learn more generalized patterns. This method strikes a balance between underfitting and overfitting, where underfitting occurs when the model is too simple to capture the underlying trends in the data, leading to both training and validation accuracy being low.

Role Of Regularization

In Python, Regularization is a technique used to prevent overfitting by adding a penalty term to the loss function, discouraging the model from assigning too much importance to individual features or coefficients.

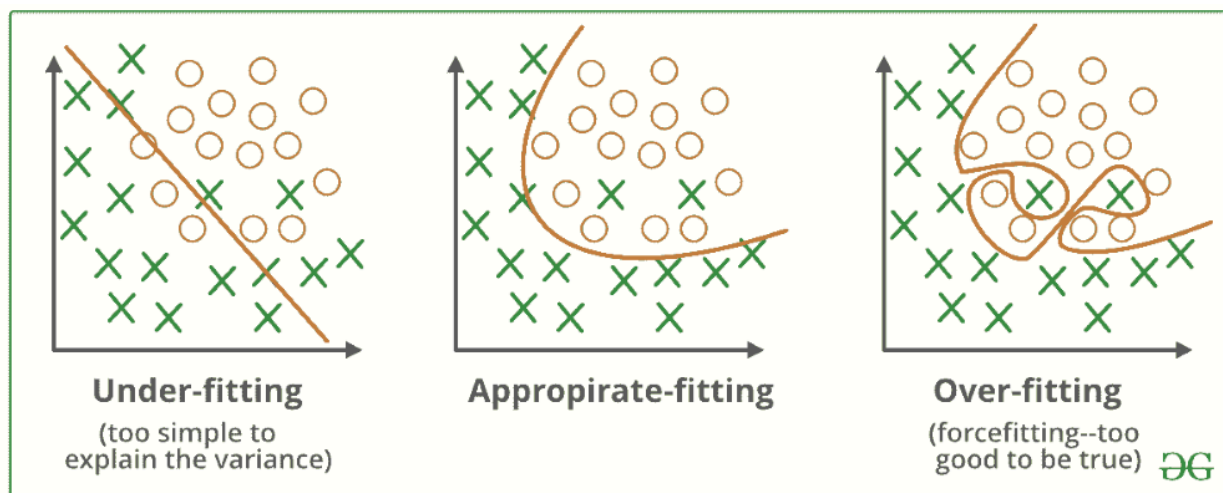
Let's explore some more detailed explanations about the role of Regularization in Python:

1. **Complexity Control:** Regularization helps control model complexity by preventing overfitting to training data, resulting in better generalization

to new data.

2. **Preventing Overfitting:** One way to prevent overfitting is to use regularization, which penalizes large coefficients and constrains their magnitudes, thereby preventing a model from becoming overly complex and memorizing the training data instead of learning its underlying patterns.
3. **Balancing Bias and Variance:** Regularization can help balance the trade-off between model bias (underfitting) and model variance (overfitting) in machine learning, which leads to improved performance.
4. **Feature Selection:** Some regularization methods, such as L1 regularization (Lasso), promote sparse solutions that drive some feature coefficients to zero. This automatically selects important features while excluding less important ones.
5. **Handling Multicollinearity:** When features are highly correlated (multicollinearity), regularization can stabilize the model by reducing coefficient sensitivity to small data changes.
6. **Generalization:** Regularized models learn underlying patterns of data for better generalization to new data, instead of memorizing specific examples.

What are Overfitting and Underfitting?



Overfitting is a phenomenon that occurs when a [Machine Learning](https://www.geeksforgeeks.org/machine-learning/) model is constrained to the training set and not able to perform well on unseen data. That is when our model learns the noise in the training data as well.

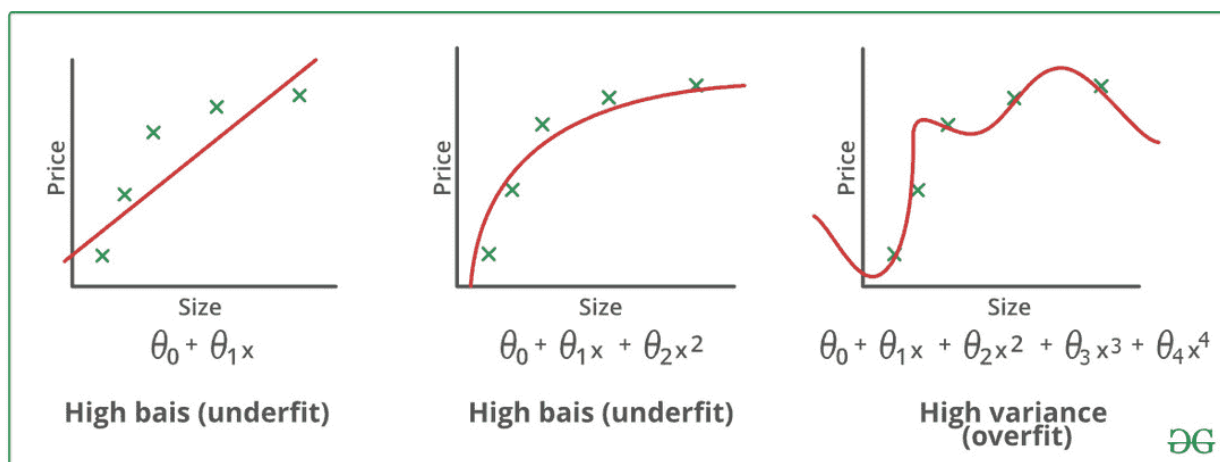
This is the case when our model memorizes the training data instead of learning the patterns in it.

Underfitting on the other hand is the case when our model is not able to learn even the basic patterns available in the dataset. In the case of the underfitting model is unable to perform well even on the training data hence we cannot expect it to perform well on the validation data. This is the case when we are supposed to increase the complexity of the model or add more features to the feature set.

What are Bias and Variance?

Bias refers to the errors which occur when we try to fit a statistical model on real-world data which does not fit perfectly well on some mathematical model. If we use a way too simplistic a model to fit the data then we are more probably face the situation of **High Bias** which refers to the case when the model is unable to learn the patterns in the data at hand and hence performs poorly.

Variance implies the error value that occurs when we try to make predictions by using data that is not previously seen by the model. There is a situation known as **high variance** that occurs when the model learns noise that is present in the data.

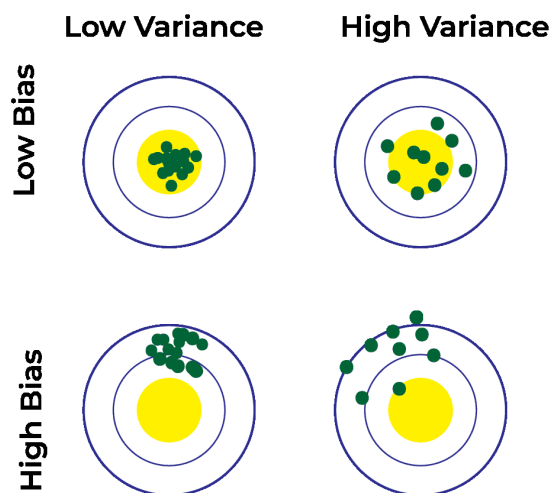


Finding a proper balance between the two that is also known as the [Bias-Variance Tradeoff](#) can help us prune the model from getting overfitted to

the training data.

Different Combinations of Bias-Variance

There can be four combinations between bias and variance:



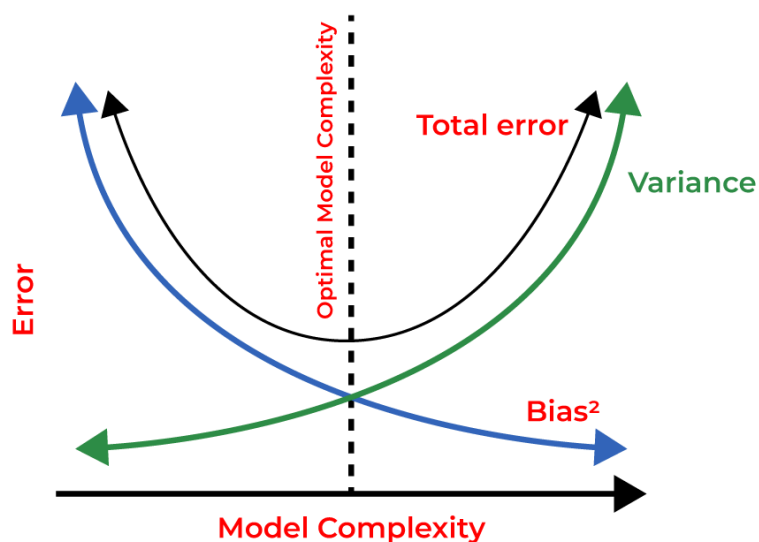
- **High Bias, Low Variance:** A model that has high bias and low variance is considered to be underfitting.
- **High Variance, Low Bias:** A model that has high variance and low bias is considered to be overfitting.
- **High-Bias, High-Variance:** A model with high bias and high variance cannot capture underlying patterns and is too sensitive to training data changes. On average, the model will generate unreliable and inconsistent predictions.
- **Low Bias, Low Variance:** A model with low bias and low variance can capture data patterns and handle variations in training data. This is the perfect scenario for a machine learning model where it can generalize well to unseen data and make consistent, accurate predictions. However, in reality, this is not feasible.

Bias Variance tradeoff

The bias-variance tradeoff is a fundamental concept in machine learning. It refers to the balance between bias and variance, which affect predictive

model performance. Finding the right tradeoff is crucial for creating models that generalize well to new data.

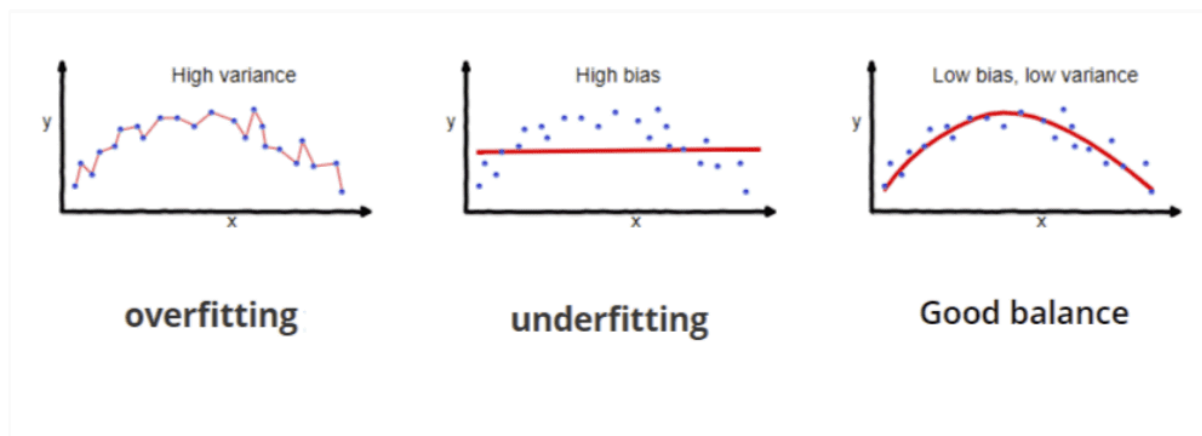
- The bias-variance tradeoff demonstrates the inverse relationship between bias and variance. When one decreases, the other tends to increase, and vice versa.
- Finding the right balance is crucial. An overly simple model with high bias won't capture the underlying patterns, while an overly complex model with high variance will fit the noise in the data.



Regularization in Machine Learning

Regularization is a technique used to reduce errors by fitting the function appropriately on the given training set and avoiding overfitting. The commonly used [regularization techniques](#) are :

1. Lasso Regularization – L1 Regularization
2. Ridge Regularization – L2 Regularization
3. Elastic Net Regularization – L1 and L2 Regularization



Lasso Regression

A regression model which uses the **L1 Regularization** technique is called **LASSO(Least Absolute Shrinkage and Selection Operator)** regression. **Lasso Regression** adds the “*absolute value of magnitude*” of the coefficient as a penalty term to the loss function(L). Lasso regression also helps us achieve feature selection by penalizing the weights to approximately equal to zero if that feature does not serve any purpose in the model.

$$\text{Cost} = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2 + \lambda \sum_{i=1}^m |w_i|$$

where,

- m – Number of Features
- n – Number of Examples
- y_i – Actual Target Value
- $y_i(\text{hat})$ – Predicted Target Value

Ridge Regression

A regression model that uses the **L2 regularization** technique is called **Ridge regression**. **Ridge regression** adds the “*squared magnitude*” of the coefficient as a penalty term to the loss function(L).

$$\text{Cost} = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2 + \lambda \sum_{i=1}^m w_i^2$$

Elastic Net Regression

This model is a combination of L1 as well as L2 regularization. That implies that we add the absolute norm of the weights as well as the squared measure of the weights. With the help of an extra [hyperparameter](#) that controls the ratio of the L1 and L2 regularization.

$$\text{Cost} = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2 + \lambda ((1 - \alpha) \sum_{i=1}^m |w_i| + \alpha \sum_{i=1}^m w_i^2)$$

Benefits of Regularization

1. Regularization improves model generalization by reducing overfitting. Regularized models learn underlying patterns, while overfit models memorize noise in training data.
2. Regularization techniques such as L1 (Lasso) L1 regularization simplifies models and improves interpretability by reducing coefficients of less important features to zero.
3. Regularization improves model performance by preventing excessive weighting of outliers or irrelevant features.
4. Regularization makes models stable across different subsets of the data. It reduces the sensitivity of model outputs to minor changes in the training set.
5. Regularization prevents models from becoming overly complex, which is especially important when dealing with limited data or noisy environments.
6. Regularization can help handle multicollinearity (high correlation between features) by reducing the magnitudes of correlated coefficients.
7. Regularization introduces hyperparameters (e.g., alpha or lambda) that control the strength of regularization. This allows fine-tuning models to achieve the right balance between bias and variance.
8. Regularization promotes consistent model performance across different datasets. It reduces the risk of dramatic performance changes when encountering new data.

Get IBM Certification and a **90% fee refund** on completing 90% course in 90 days! [Take the Three 90 Challenge today.](#)

Master Machine Learning, Data Science & AI with this complete program and also get a 90% refund. What more motivation do you need? [Start the challenge right away!](#)

[Data Science IBM Certification](#)[Data Science](#)[Data Science Projects](#)[Data Analysis](#)[Data Visualization](#)[Ma](#)[Comment](#)[More info](#)[Advertise with us](#)

Next Article

Regularization Techniques in
Machine Learning

Similar Reads

Regularization Techniques in Machine Learning

Overfitting is a major concern in the field of machine learning, as models aim to extract complex patterns from data. When a model learns to commit the...

10 min read

Dropout Regularization in Deep Learning

Training a model excessively on available data can lead to overfitting, causing poor performance on new test data. Dropout regularization is a...

4 min read

Getting started with Machine Learning || Machine Learning Roadmap

Machine Learning (ML) represents a branch of artificial intelligence (AI) focused on enabling systems to learn from data, uncover patterns, and...

11 min read

How L1 Regularization brings Sparsity`

Prerequisites: Regularization in ML We know that we use regularization to avoid underfitting and over fitting while training our Machine Learning...

3 min read

Regularization by Early Stopping

Regularization is a kind of regression where the learning algorithms are modified, to reduce overfitting. This may incur a higher bias but will lead to...

4 min read

LightGBM Regularization parameters

LightGBM is a powerful gradient-boosting framework that has gained immense popularity in the field of machine learning and data science. It is...

11 min read

Overfitting and Regularization in ML

The effectiveness of a machine learning model is measured by its ability to make accurate predictions and minimize prediction errors. An ideal or good...

14 min read

CatBoost Regularization parameters

CatBoost, developed by Yandex, is a powerful open-source gradient boosting library designed to tackle categorical feature handling and deliver...

9 min read

How does L1 and L2 regularization prevent overfitting?

Overfitting is a recurring problem in machine learning that can harm a model's capacity to perform well and be generalized. Regularization is a...

4 min read

ML | Implementing L1 and L2 regularization using Sklearn

Prerequisites: L2 and L1 regularization This article aims to implement the L2 and L1 regularization for Linear regression using the Ridge and Lasso...

3 min read

**Corporate & Communications Address:**

A-143, 7th Floor, Sovereign Corporate Tower, Sector- 136, Noida, Uttar Pradesh (201305)

Registered Address:

K 061, Tower K, Gulshan Vivante Apartment, Sector 137, Noida, Gautam Buddh Nagar, Uttar Pradesh, 201305



Advertise with us

Company

About Us
Legal
Privacy Policy
Careers
In Media
Contact Us
Advertise with us
GFG Corporate Solution
Placement Training Program

Languages

Python
Java
C++
PHP
GoLang
SQL
R Language
Android Tutorial

Data Science & ML

Data Science With Python

Explore

Job-A-Thon Hiring Challenge
Hack-A-Thon
GfG Weekly Contest
Offline Classes (Delhi/NCR)
DSA in JAVA/C++
Master System Design
Master CP
GeeksforGeeks Videos
Geeks Community

DSA

Data Structures
Algorithms
DSA for Beginners
Basic DSA Problems
DSA Roadmap
DSA Interview Questions
Competitive Programming

Web Technologies

HTML

Data Science For Beginner

Machine Learning

ML Maths

Data Visualisation

Pandas

NumPy

NLP

Deep Learning

CSS

JavaScript

TypeScript

ReactJS

NextJS

NodeJs

Bootstrap

Tailwind CSS

Python Tutorial

Python Programming Examples

Django Tutorial

Python Projects

Python Tkinter

Web Scraping

OpenCV Tutorial

Python Interview Question

DevOps

Git

AWS

Docker

Kubernetes

Azure

GCP

DevOps Roadmap

School Subjects

Mathematics

Physics

Chemistry

Biology

Social Science

English Grammar

Databases

SQL

MYSQL

PostgreSQL

PL/SQL

MongoDB

Computer Science

GATE CS Notes

Operating Systems

Computer Network

Database Management System

Software Engineering

Digital Logic Design

Engineering Maths

System Design

High Level Design

Low Level Design

UML Diagrams

Interview Guide

Design Patterns

OOAD

System Design Bootcamp

Interview Questions

Commerce

Accountancy

Business Studies

Economics

Management

HR Management

Finance

Income Tax

Preparation Corner

Company-Wise Recruitment Process

Resume Templates

Aptitude Preparation

Puzzles

Company-Wise Preparation

Companies

Colleges

Competitive Exams

JEE Advanced
UGC NET
UPSC
SSC CGL
SBI PO
SBI Clerk
IBPS PO
IBPS Clerk

Free Online Tools

Typing Test
Image Editor
Code Formatters
Code Converters
Currency Converter
Random Number Generator
Random Password Generator

DSA/Placements

DSA - Self Paced Course
DSA in JavaScript - Self Paced Course
DSA in Python - Self Paced
C Programming Course Online - Learn C with Data Structures
Complete Interview Preparation
Master Competitive Programming
Core CS Subject for Interview Preparation
Mastering System Design: LLD to HLD
Tech Interview 101 - From DSA to System Design [LIVE]
DSA to Development [HYBRID]
Placement Preparation Crash Course [LIVE]

Machine Learning/Data Science

Complete Machine Learning & Data Science Program - [LIVE]
Data Analytics Training using Excel, SQL, Python & PowerBI - [LIVE]
Data Science Training Program - [LIVE]
Mastering Generative AI and ChatGPT
Data Science Course with IBM Certification

Clouds/Devops

DevOps Engineering
AWS Solutions Architect Certification
Salesforce Certified Administrator Course

More Tutorials

Software Development
Software Testing
Product Management
Project Management
Linux
Excel
All Cheat Sheets
Recent Articles

Write & Earn

Write an Article
Improve an Article
Pick Topics to Write
Share your Experiences
Internships

Development/Testing

JavaScript Full Course
React JS Course
React Native Course
Django Web Development Course
Complete Bootstrap Course
Full Stack Development - [LIVE]
JAVA Backend Development - [LIVE]
Complete Software Testing Course [LIVE]
Android Mastery with Kotlin [LIVE]

Programming Languages

C Programming with Data Structures
C++ Programming Course
Java Programming Course
Python Full Course

GATE

GATE CS & IT Test Series - 2025
GATE DA Test Series 2025
GATE CS & IT Course - 2025
GATE DA Course 2025

