Session - 2

MODEL TRAINING

Linear regression is one of the fundamental statistical and machine learning techniques. Whether you want to do statistics, machine learning, or scientific computing, there's a good chance that you'll need it. It's best to build a solid foundation first and then proceed toward more complex methods.

What students learn?

- What linear regression **is**
- What linear regression is **used** for
- How linear regression **works**
- How to **implement** linear regression in Python, step by step

What is Model Training?

Model training is the phase in the data science development lifecycle where practitioners try to fit the best combination of weights and bias to a machine learning algorithm to minimize a loss function over the prediction range. The p urp ose of model training is to build the best mathematical representation of the relationship between data features and a target label (in supervised learning) or among the features themselves (unsupervised learning). Loss functions are a critical aspect of model training since they define how to optimize the machine learning algorithms. Depending on the objective, type of data and algorithm, data science practitioner use different type of loss functions. One of the popular examples of loss functions is Mean Square Error (MSE).

Why is it Important?

Model t raining is the key step in machine learning that results in a model ready to be validated, tested, and deployed. The performance of the model determines the quality of the applications that are built using it. Quality of training data and the training algorithm are both important assets during the model training phase. Typically, training data is split for training, validation and testing. The training algorithm is chosen based on the end use case. There are a number of tradeoff points in deciding the best algorithm–model complexity, interpretability, performance, compute requirements, etc. All these aspects of model training make it both an involved and important process in the overall machine learning development cycle.

What is Supervised Machine Learning?

Supervised Machine Learning refers to a method of developing a predictive function by using a training set of labeled examples that pair input data with labeled output. Once the function optimizes how it associates certain input values with labeled outputs, the function can be tested with additional data to validate its accuracy in predicting the right label. With sufficient accuracy levels from those training and validation runs, the function can then be applied operationally to new input data to create output labels. It is important to measure, monitor, and adjust the predictive function over time as input data may shift or the algorithm may have an opportunity to improve its accuracy using a longer history, additional inputs, or a different approach.

Why is Supervised Machine Learning important?

Supervised Machine Learning is important for automating tasks that need to be performed at a scale or speed that is too challenging or expensive for humans to perform, particularly for use cases like image recognition and speech translation. Creating an effective predictive function depends on having a full set of representative data that is accurately labeled, as well as labels that can be sufficiently differentiated using the available data.

Regression

Regression analysis is one of the most important fields in statistics and machine learning. There are many regression methods available. Linear regression is one of them.

What Is Regression?

Regression searches for relationships among **variables**. For example, you can observe several employees of some company and try to understand how their salaries depend on their **features**, such as experience, education level, role, city of employment, and so on.

This is a regression problem where data related to each employee represents one **observation**. The presumption is that the experience, education, role, and city are the independent features, while the salary depends on them.

Similarly, you can try to establish the mathematical dependence of housing prices on area, number of bedrooms, distance to the city center, and so on.

Generally, in regression analysis, you consider some phenomenon of interest and have a number of observations. Each observation has two or more features. Following the assumption that at least one of the features depends on the others, you try to establish a relation among them.

In other words, you need to find a **function that maps some features or variables to others** sufficiently well.

The dependent features are called the **dependent variables**, **outputs**, or **responses**. The independent features are called the **independent variables**, **inputs**, **regressors**, or **predictors**.

Regression problems usually have one continuous and unbounded dependent variable. The inputs, however, can be continuous, discrete, or even categorical data such as gender, nationality, or brand.

It's a common practice to denote the outputs with y and the inputs with x. If there are two or more independent variables, then they can be represented as the vector $\mathbf{x} = (x_1, \ldots, x_r)$, where r is the number of inputs.

When Do You Need Regression?

Typically, you need regression to answer whether and how some phenomenon influences the other or how several variables are related. For example, you can use it to determine *if* and *to what extent* experience or gender impacts salaries.

Regression is also useful when you want to forecast a response using a new set of predictors. For example, you could try to predict electricity consumption of a household for the next hour given the outdoor temperature, time of day, and number of residents in that household.

Regression is used in many different fields, including economics, computer science, and the social sciences. Its importance rises every day with the availability of large amounts of data and increased awareness of the practical value of data.

What is Regularization in Machine Learning?

Regularization refers to techniques that are used to calibrate machine learning models in order to minimize the adjusted loss function and prevent overfitting or underfitting.

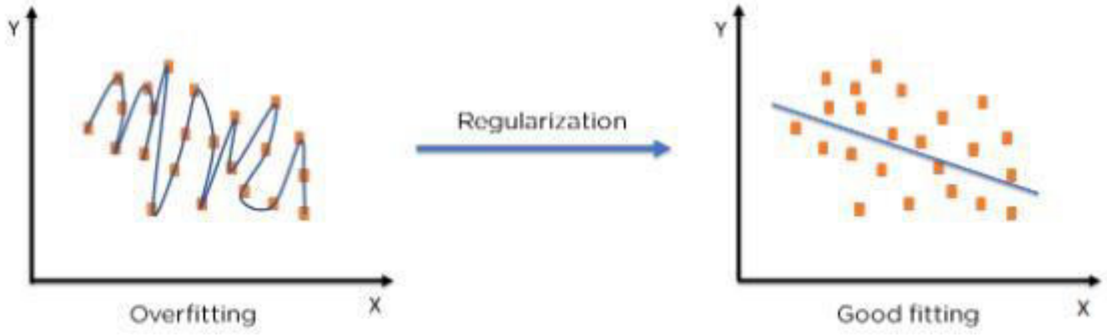


Figure 1: Regularization on an over-fitted model

Using Regularization, we can fit our machine learning model appropriately on a given test set and hence reduce the errors in it.

Regularization Techniques

There are two main types of regularization techniques: Ridge Regularization and Lasso Regularization.

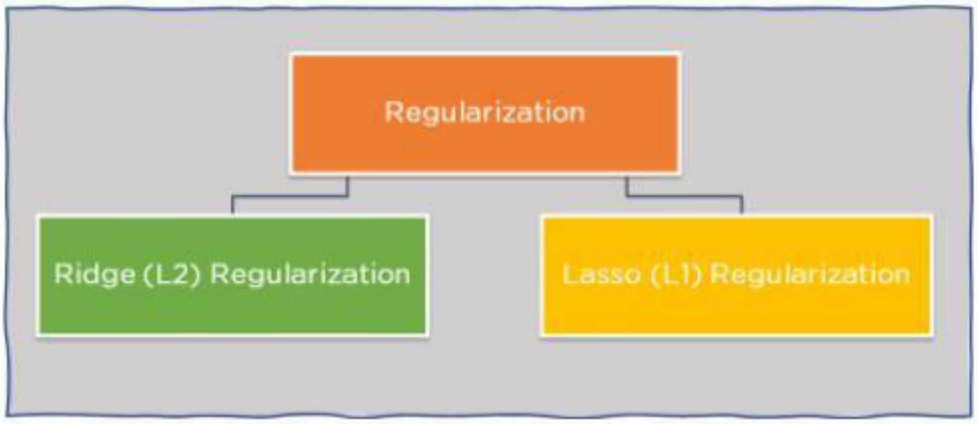
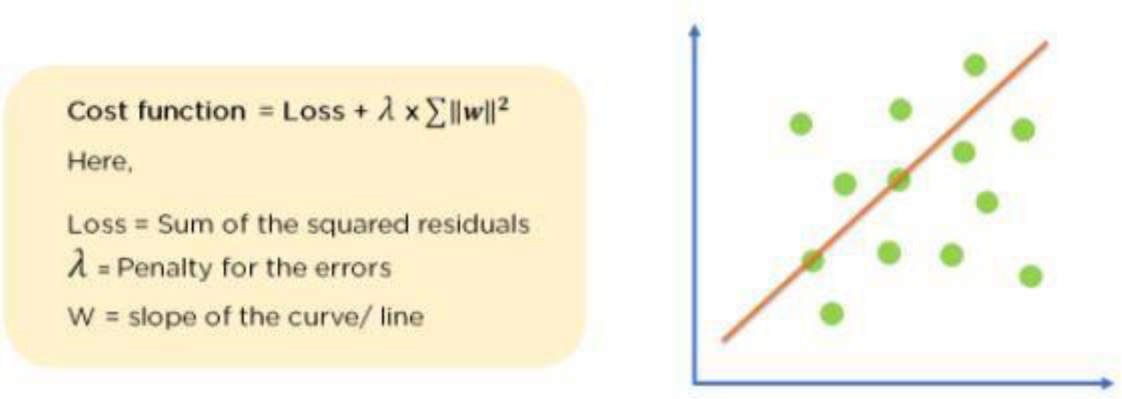


Figure 2: Regularization techniques

Ridge Regularization :

Also known as Ridge Regression, it modifies the over-fitted or under fitted models by adding the penalty equivalent to the sum of the squares of the magnitude of coefficients.

This means that the mathematical function representing our machine learning model is minimized and coefficients are calculated. The magnitude of coefficients is squared and added. Ridge Regression performs regularization by shrinking the coefficients present. The function depicted below shows the cost function of ridge regression :



Cost function = Loss + λ x $\sum \|w\|^2$

Here,

Loss = Sum of the squared residuals

λ = Penalty for the errors

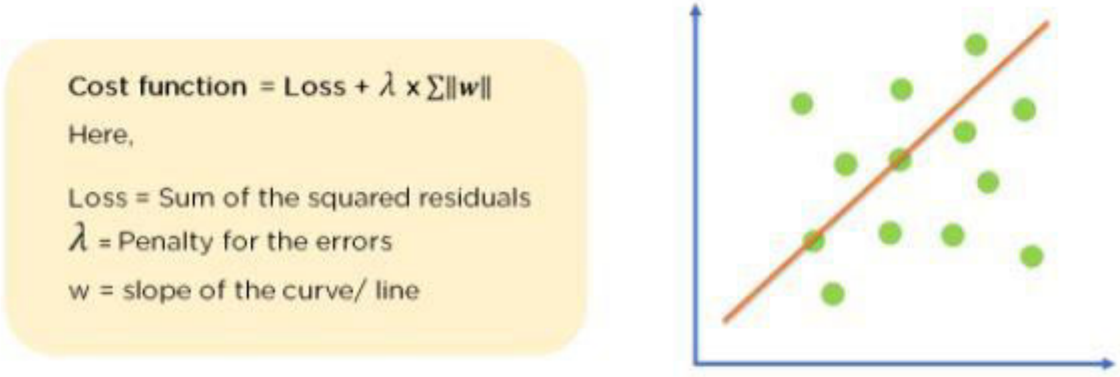
W = slope of the curve/ line

Figure 3: Regularization on an over-fitted model

Lasso Regularization

It modifies the over-fitted or under-fitted models by adding the penalty equivalent to the sum of the absolute values of coefficients.

Lasso regression also performs coefficient minimization, but instead of squaring the magnitudes of the coefficients, it takes the true values of coefficients. This means that the coefficient sum can also be 0, because of the presence of negative coefficients. Consider the cost function for Lasso regression:



Cost function = Loss + λ x $\sum \|w\|$

Here,

Loss = Sum of the squared residuals

λ = Penalty for the errors

w = slope of the curve/ line

Figure 4: Cost function for Lasso Regression

Real-world examples

The following represents some real-world examples / use cases where linear regression models can be used:

- **Forecasting sales**: Organizations often use linear regression models to forecast future sales. This can be helpful for things like budgeting and planning. Algorithms such as Amazon’s item-to-item collaborative filtering are used to predict what customers will buy in the future based on their past purchase history.
- **Cash forecasting**: Many businesses use linear regression to forecast how much cash they’ll have on hand in the future. This is important for things like managing expenses and ensuring that there is enough cash on hand to cover unexpected costs.
- **Analyzing survey data**: Linear regression can also be used to analyze survey data. This can help businesses understand things like customer satisfaction and product preferences. For example, a company might use linear regression to figure out how likely people are to recommend their product to others.
- **Stock predictions**: A lot of businesses use linear regression models to predict how stocks will perform in the future. This is done by analyzing past data on stock prices and trends to identify patterns.
- **Predicting consumer behavior**: Businesses can use linear regression to predict things like how much a customer is likely to spend. Regression models can also be used to predict consumer behavior. This can be helpful for things like targeted marketing and product development. For example, Walmart uses linear regression to predict what products will be popular in different regions of the country.
- **Analysis of relationship between variables**: Linear regression can also be used to identify relationships between different variables. For example, you could use linear regression to find out how temperature affects ice cream sales.

Linear Regression

Linear regression is probably one of the most important and widely used regression techniques. It’s among the simplest regression methods. One of its main advantages is the ease of interpreting results.

What is Linear Regression?

Linear regression is a machine learning concept that is used to build or train the models (mathematical models or equations) for solving **supervised learning problems** related to predicting **continuous numerical value.** Supervised learning problems represent the class of the problems where the value (data) of the independent or predictor variable (features) and the dependent or response variables are already known. The known values of the dependent and independent variable (s) are used to come up with a mathematical model/formula which is later used to predict / estimate output given the value of input features. In natural and social sciences problems, linear regression is used to determine the relationship between input and output variables. In machine learning tasks, linear regression is mostly used for making the prediction of numerical values from a set of input values.

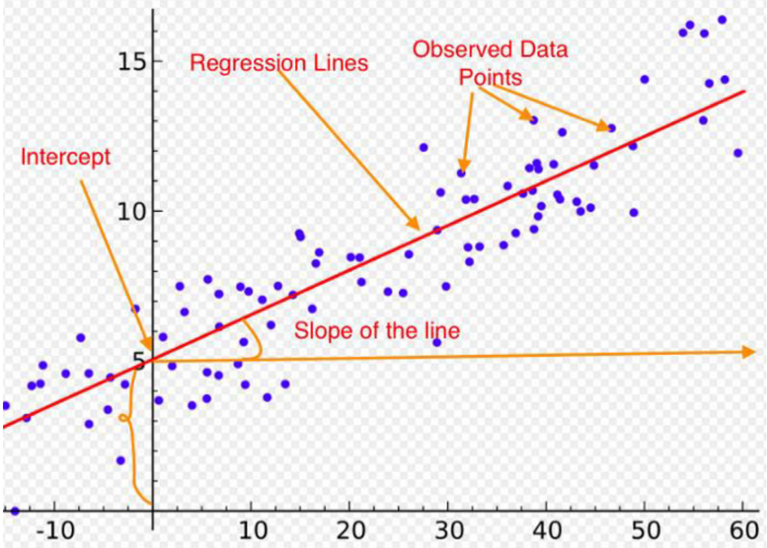


Figure 5: Regularization on an over-fitted model

The linear regression mathematical structure or model assumes that there is a linear relationship between input and output variables. In addition, it is also assumed that the noise or error is well-mannered (normal or Gaussian distribution). Building linear regression models represents determining the value of **output (dependent/response variable)** as a function of **the weighted sum of input features (independent / predictor variables)**. This data is used to determine the most optimum value of the coefficients of the independent variables.

Let's say, there is a numerical response variable, Y, and one or more predictor variables X1, X2, etc. And, there is some relationship between Y and X that can be written as the following:

Y = f(X) + error

Where f is some fixed but unknown function of X1 and X2. **When the unknown function is a linear function of X1 and X2, the Y becomes a linear regression function** or model such as the following. Note that the error term averages out to be zero.

Y = b0 + b1*X1 + b2*X2

In the above equation, different values of Y and X1, and X2 are known during the model training phase. As part of training the model, the most optimal value of coefficients b1, b2, and b0 are determined based on the least square regression algorithm. The **least-squares method** is an algorithm to find the best fit for a set of data points by minimizing the sum of the squared residuals or square of error of points (actual values representing the response variable) from the points on the plotted curve (predicted value). This is shown below.

If YiYi is the ith observed value and Yi^Yi^ is the ith response value, then the ith residual or error value is calculated as the following:

$$e_i = Y_i - \hat{Y_i}$$

The residual sum of squares can then be calculated as the following:

$$RSS = {e_1}^2 + {e_2}^2 + {e_3}^2 + \ldots + {e_n}^2$$

In order to come up with the optimal linear regression model, the least-squares method as discussed above represents minimizing the value of RSS (Residual sum of squares).

Different types of linear regression models

There are two different types of linear regression models. They are the following:

- **Simple linear regression**: The following represents the simple linear regression where there is just one independent variable, X, which is used to predict the dependent variable Y.

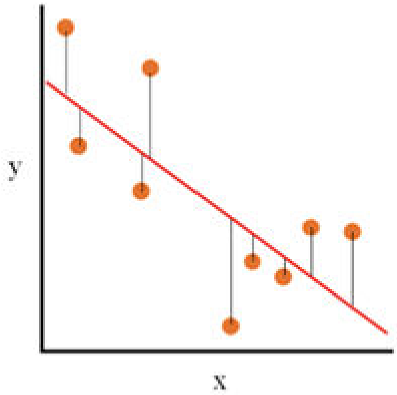


Fig 7. Simple linear regression

- **Multiple linear regression**: The following represents the multiple linear regression where there are two or more independent variables (X1, X2) that are used for predicting the dependent variable Y.

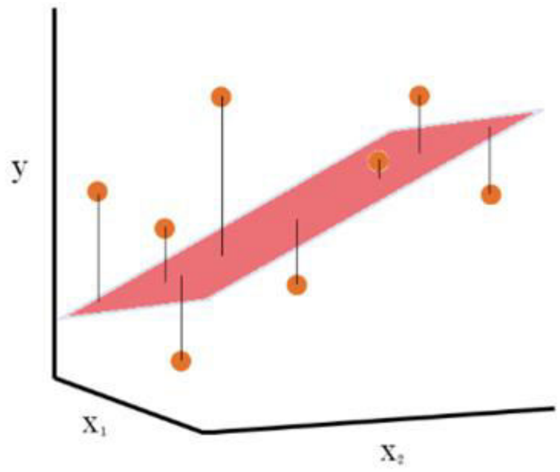


Fig 8. Multiple linear regression

Polynomial Regression

You can regard polynomial regression as a generalized case of linear regression. You assume the polynomial dependence between the output and inputs and, consequently, the polynomial estimated regression function.

In other words, in addition to linear terms like b_1x_1, your regression function f can include nonlinear terms such as $b_2x_1^2$, $b_3x_1^3$, or even $b_4x_1x_2$, $b_5x_1^2x_2$.

The simplest example of polynomial regression has a single independent variable, and the estimated regression function is a polynomial of degree two: $(x) = b_0 + b_1x + b_2x^2$.

Now, remember that you want to calculate b_0, b_1, and b_2 to minimize SSR. These are your unknowns!

Keeping this in mind, compare the previous regression function with the function $(x_1, x_2) = b_0 + b_1x_1 + b_2x_2$, used for linear regression. They look very similar and are both linear functions of the unknowns b_0, b_1, and b_2. This is why you can solve the **polynomial regression problem** as a **linear problem** with the term x^2 regarded as an input variable.

In the case of two variables and the polynomial of degree two, the regression function has this form: $(x_1, x_2) = b_0 + b_1x_1 + b_2x_2 + b_3x_1^2 + b_4x_1x_2 + b_5x_2^2$.

The procedure for solving the problem is identical to the previous case. You apply linear regression for five inputs: x_1, x_2, x_1^2, x_1x_2, and x_2^2. As the result of regression, you get the values of six weights that minimize SSR: b_0, b_1, b_2, b_3, b_4, and b_5.

Simple Linear Regression Example

As shown above, simple linear regression models comprise of one input feature (independent variable) which is used to predict the value of the output (dependent) variable. The following mathematical formula represents the regression model:

Y = b*X + b0

Let's take an example comprising one input variable used to predict the output variable. However, in real life, it may get difficult to find a supervised learning problem that could be modeled using simple linear regression.

Simple Linear Model for Predicting Marks

Let's consider the problem of predicting the marks of a student based on the number of hours he/she put into the preparation. Although at the outset, it may look like a problem that can be modeled using simple linear regression, it could turn out to be a multiple linear regression problem depending on multiple input features. Alternatively, it may also turn out to be a non-linear problem. However, for the sake of example, let's consider this as a simple linear regression problem.

However, let's assume for the sake of understanding that the marks of a student (M) do depend on the number of hours (H) he/she put up for preparation. The following formula can represent the model:

Marks = function (No. of hours)

=> Marks = m*Hours + c

The best way to determine whether it is a simple linear regression problem is to do a **plot of Marks vs Hours**. If the plot comes like below, it may be inferred that a linear model can be used for this problem.

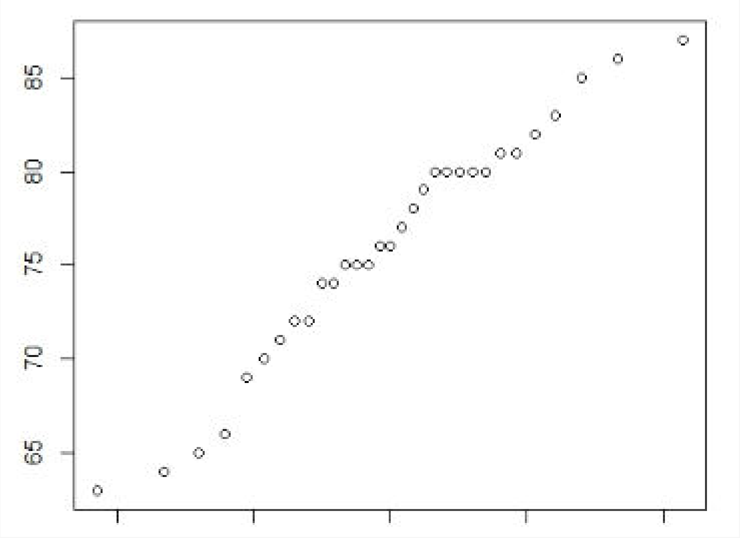


Fig 9. Plot representing a simple linear model for predicting marks

The data represented in the above plot would be used to find out a line such as the following which represents a best-fit line. The slope of the best-fit line would be the value of "m".

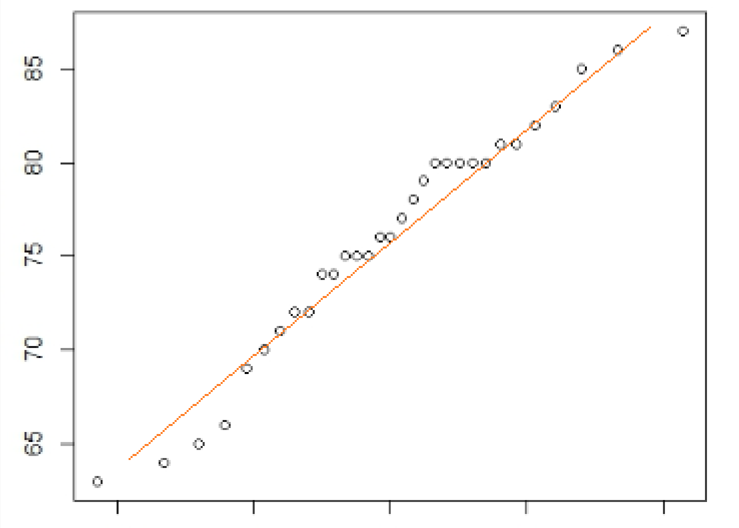


Fig 10. Plot representing a simple linear model with a regression line

The value of **m (slope of the line)** can be determined using an **objective function** which is a combination of the **loss function** and **a regularization term**. For simple linear regression, the objective function would be the summation of Mean Squared Error (MSE). MSE is the sum of squared distances between the target variable (actual marks) and the predicted values (marks calculated using the above equation). The best fit line would be obtained by **minimizing the objective function (summation of mean squared error).**

Multiple Linear Regression Example

Multiple linear regression can be used to model the supervised learning problems where there are two or more input (independent) features that are used to predict the output variable. The following formula can be used to represent a typical multiple regression model:

Y = b0 + b1*X1 + b2*X2 + b3*X3 + … + bn*Xn

In the above example, Y represents the response/dependent variable, and X1, X2, and X3 represent the input features. The model (mathematical formula) is trained using training data to find the optimum values of b0, b1, b2, and b3 which minimizes the **objective function (mean squared error)**.