Linear Regression and Logistic Regression

*What is supervised Learning?*

It’s a type of machine learning where the algorithm learns from labelled data. Labelled data means the dataset whose respective target value is already known. It has two types Classification and regression. Classification predicts the class of the dataset based on independent input variable. Regression predicts the continuous output variables based on the independent input variable. Like the prediction of house prices based on different parameters like house age, distance from the main road, location, area etc. Eg of supervised learning: linear regression, logistic regression, k nearest neighbour, svm, decision trees, random forest, neural networks. Eg of unsupervised learning are: k means clustering, PCA, DBSCAN

**Linear Regression**

Supervised Machine learning algorithm that computes the linear relationship between the dependent variable and one or more independent features by fitting a linear equation to observed data.

When there is only one independent feature, its called simple linear regression and when there are more than one features its called multiple linear regression.

When there is only one dependent variable its called univariate linear regression and when there are more than one dependent variables its called multivariate regression

Simple Linear Regression

Involves only one independent variable and one dependent variable. The equation for simple linear regression is

*y*=*β*0​+*β*1​*X*

Y is the dependent variable

B0 is the intercept(bias)

B1 is the slope(weights)

X is the independent variable

Linear regressions primary objective is to locate the best fit line which implies the error between the predicted value and the actual value should be kept minimum. The least error would be there in the best fit line. The slope of the line indicates how much the dependent variable changes with a unit change in the independent variable

To achieve the best fit regression line the model aims to predict the target value such that the error difference between the predicted value and the true value is minimum. So its required to update B0, B1 to reach the best value that minimizes the error between the predicted value and the true value.   
  
In regression, the difference between the observed value of the dependent variable and the predicted value is called the residuals.

The best fit line is obtained by minimizing the sum of squared residuals. But for gradient descent we actually minimize the mean squared error(sum of squared residuals divided by the total number of values) because adding a factor of 1/n would make updates more consistent and gradient descent would be more stable. Whereas just using RSS would mean bigger updates thus making gradient descent unstable. Although you can implement linear regression with both, minimizing MSE is preferred.

*Minimize* 1/n​∑(1 to n)​(*yi*​^​−*yi*​)2

Cost Function

Cost function means the average of the loss function

In linear regression, Mean Square error cost function is used, which calculates the average of the squared error between the predicted values and the actual values. The purpose is to determine the optimal values for B0 and B1 providing the best fit line for the given data points

Cost function(*J*)=*1/n*​∑*ni*​(*yi*​^​−*yi*​)2

Utilizing the MSE function the iterative process of gradient descent is applied. This ensures that the MSE value converges to the global minima, signifying the most accurate fit of the linear regression line to the dataset. It involves continuously adjusting the parameters based on the gradients calculated from the MSE. Final result is a linear regression line that minimizes the overall squared differences between the predicted and actual values.

Gradient descent

This algorithm is used to train a linear regression model by iteratively modifying the models parameters to reduce the mean square error of the model on a training dataset. The idea is to start with random values for B0 and B1 and then iteratively update the values reaching minimum cost.

A black background with white text

Description automatically generated

A black background with white text

Description automatically generated

Parameter Updates

A group of math equations

Description automatically generated

Here alpha is the learning rate. It decides how fast the algorithm converges to the minima.

A diagram of learning and low learning

Description automatically generated

Assumptions of Simple Linear Regression

1. Linearity: The independent and dependent variables have a linear relationship with one another. This implies that changes in the dependent variable follow those in the independent variables in a linear fashion. It means that there should be a straight line that can be drawn through the data points. If the relationship is not linear then the model wont be accurate.
2. Independence: Observations in the dataset are independent of each other. Value of dependent variable of 1 observation shouldn’t affect the value of the dependent variable in another observation.
3. Homoscedasticity: Errors are spread evenly across the predictions For Eg: If we are predicting house prices the error for small houses and large houses should not be a very large value
4. Normality: Residuals should be normally distributed, that is it should follow a bell shaped curve

Assumptions of Multiple Linear Regression

1. No multicollinearity :Multicollinearity is a statistical phenomenon where two or more independent variables are highly correlated making it difficult to asses the individual effects of each variable on the independent variable. To detect multicollinearity we find out the correlation matrix. High correlation (close to 1 or -1) indicate multicollinearity. Variance inflation factor: It’s a measure of how much the regression coefficient increases if the predictors are correlated. A high VIF means multicollinearity.
2. Feature Selection: its essential to carefully select the features in a model otherwise including redundant features would lead to overfitting.

Evaluation Metrics

1. Mean Square Error: It calculates the average of the squared differences between the predicted values and the actual values. It is squared to make sure that negative and positive differences aren’t cancelled out.

MSE = 1/n ∑(y\_predicted – y)^2

Its sensitive to outliers as large errors contribute significantly to the overall score

1. Mean absolute Error : 1/n ∑(y\_actual – y\_predicted). Lower MAE indicates better model performance. It is not sensitive to outliers as we consider absolute differences
2. Root Mean Square: Square root of MSE. Not such a good metric better metrics like R-squared are used
3. R-Squared(Coefficient of determination) :

R^2 = 1 – (RSS/ TSS)

Where RSS is residual sum of squares and TSS is total sum of squares. Total sum of squares is the square of the differences between the data points and the mean

Bias

It refers to how far off the models predictions are from the true value resulting from a very simple model. It’s a measure of how far off your models predictions are on average from the underlying relationship in the data.

High Bias = Underfitting. To fix high bias we usually increase model complexity

Variance

It refers to how much the models predictions change when the dataset is changed. A model with high variance captures noise also thus performing poorly on unseen data. High variance = Overfitting. To fix high variance we reduce model complexity by using regularization, reducing the number of features or just gathering more data

Bias Variance Trade off

You cannot minimize both simultaneously. A balanced should be maintained

A diagram of a model complex

Description automatically generated with medium confidence

What is Overfitting?

When the model performs really well on training data but performs poorly on testing data. This is when we say the model has overfit.

Steps to prevent overfitting: 1) cross validation, regularization

Underfitting

When the model fails to learn from the training dataset, that it is not able to generalize on the testing dataset. It leads to low training and testing accuracy. Steps to remove underfitting: increase model complexity, increase the number of features

Regularization

L1 Regularization (Lasso Regularization)

When we train a linear regression model on **complex data** with lots of features, the model may try to fit **every small detail or noise** in the training set. This leads to **overfitting**, where the model performs very well on the training data but poorly on new data (test set).

Lasso adds a penalty to large coefficients. If a features coefficient is too large, lasso will shrink it thus reducing the model complexity thus also reducing overfitting by removing irrelevant features. It performs feature selection by completely shrinking their coefficients to 0

Lasso Cost Function=RSS + λ∑​∣βj​

Lambda here is the regularization parameter. A small value means the model will keep most of the features and a large value means the model will shrink more 6yfjk;’ coefficients to 02460

Lasso regression should be avoided when all the features are important.

Pros of lasso :

Performs feature selection thus making the model simple and more interpretable. Its also good for high dimensional data

Cons:

Struggles with highly correlated features and may underperform on small datasets

Ridge Regression

Just like lasso regression a penalty term is added to the cost function. Its pretty much the same as lasso regression but here the coefficients are never shrunk to 0, they are just reduced. Ridge regression is used when we don’t want to eliminate all the features but just reduce their impact.

Ridge Cost Function=RSS+λ∑​βj2

The main aim is the same i.e to reduce overfitting by reducing model complexity​

Elastic Net regression

Elastic net regression combines the best of both l1 and l2 regularization.

A black background with white text

Description automatically generated

Like lasso it can reduce some coefficients to 0 thus eliminating those features. Like ridge regression it handles multicollinearity. It performs well when the features are highly corelated . It finds a middle ground between ridge and lasso by using both penalties

If lambda one = 0 and lambda 2 > 0, then it works as ridge regression

If lamda one > 0 and lambda 2 = 0 then it works as lasso regression

Logistic Regression

Logistic Regression is another statistical analysis method borrowed by Machine Learning. It is used when our dependent variable is dichotomous or binary. It just means a variable that has only 2 outputs, for example, A person will survive this accident or not, The student will pass this exam or not. The outcome can either be yes or no (2 outputs). This regression technique is similar to linear regression and can be used to predict the Probabilities for classification problems.

Types of logistic regression

1. Binary Logistic regression

Used to predict the probability of a binary outcome like a yes or no problem.

1. Multinomial logistic regression

Used to predict probabilities of all the possible outcomes and the outcome with the highest possible probability is the final prediction. A softmax function is used for this regression

Why can we just use linear Regression ?

Linear regression is sensitive to outliers. Another problem with linear regression is that the predicted values may be out of range. We know that probability can be between 0 and 1, but if we use linear regression, this probability may exceed 1 or go below 0. To overcome these problems, we use Logistic Regression, which converts this straight best-fit line in linear regression to an S-curve using the sigmoid function, which will always give values between 0 and 1.

How does logistic Regression work?

So lets start with our linear regression equation



Let’s say instead of y we are taking probabilities (P). But there is an issue here, the value of (P) will exceed 1 or go below 0 and we know that range of Probability is (0-1). To overcome this issue we take ***“odds”*** of P

A black and white image of a number

Description automatically generated

The problem here is that the range is restricted and we don’t want a restricted range because if we do so then our correlation will decrease. By restricting the range we are actually decreasing the number of data points and of course, if we decrease our data points, our correlation will decrease. It is difficult to model a variable that has a restricted range. To control this we take the ***log of odds***which has a range from (-∞,+∞).

A mathematical equation with numbers and symbols

Description automatically generated

A math equations on a white background

Description automatically generated

This is how we obtain the sigmoid function. Now we use this sigmoid function to make our predictions. We can set a threshold value (like 0.5) to classify which class the value computed by the sigmoid function for that observation belongs to. The cost function is then minimized during training. We then accordingly update the weights and biases by computing the gradients of the cost function and this continues for n number of iterations till we get the final weights and biases.

With the final weights and biases we then can use the model to make predictions

Why do we use Cross Entropy Loss Function?

First lets answer the question why not mean square error? So we don’t use mean square error because the sigmoid function is non linear and upon substituting it into the mean square error loss function, we get a convex graph with many local minimas. So gradient descent would converge on a local minima which is not a desirable result. In order to solve this problem, we use another loss function called log loss which is also derived from the maximum likelihood estimator

A close up of a text

Description automatically generated

A graph of cost function and a graph of cost function

Description automatically generated

On combining both the graphs we get a graph which has only 1 minimum and so it will be easy to use gradient descent here

A graph of a function

Description automatically generated

The gradient descent algorithm is applied on this graph and accordingly we get the weights and biases.