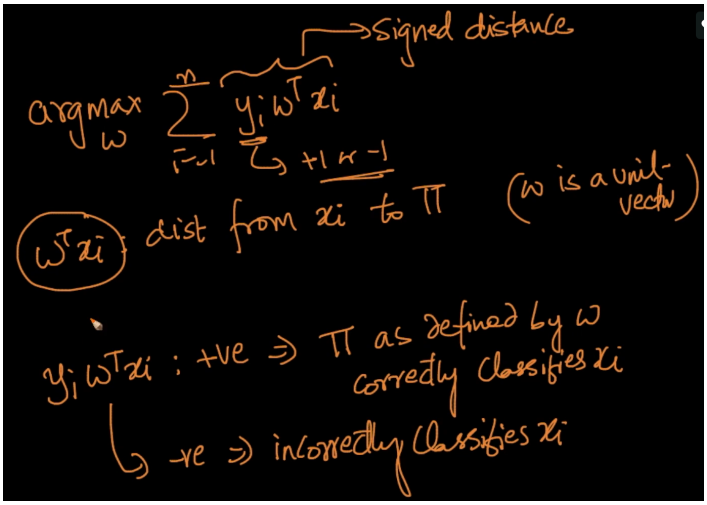
Logistic Regression

Logistic Regression is a classification algorithm. It is used to predict a binary outcome (1 / 0, Yes / No, True / False) given a set of independent variables.

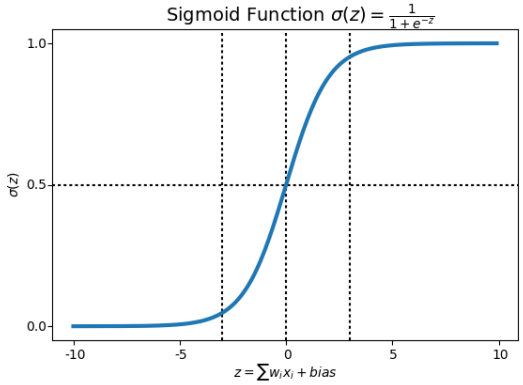
Logistic regression as a special case of Linear regression when the outcome variable is categorical, where we are using log of odds as dependent variable. In simple words, it predicts the probability of occurrence of an event by fitting data to a logit function.



Here we will always trying to maximize signed-distance

Sigmoid Function

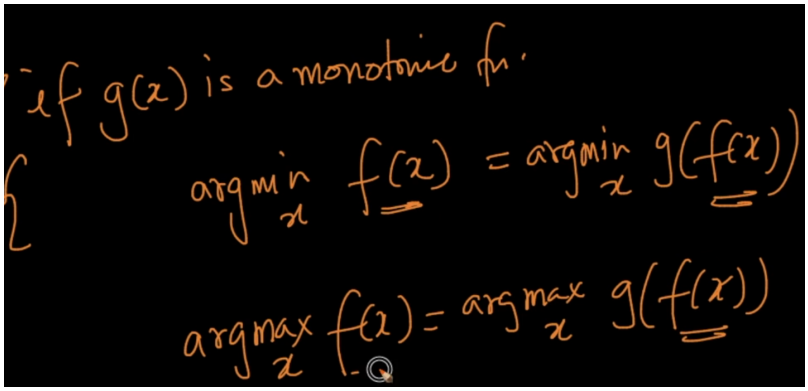
- Used to map predicted values to probabilities.
- The function maps any real value into another value between 0 and 1.
- In machine learning, we use sigmoid to map predictions to probabilities.



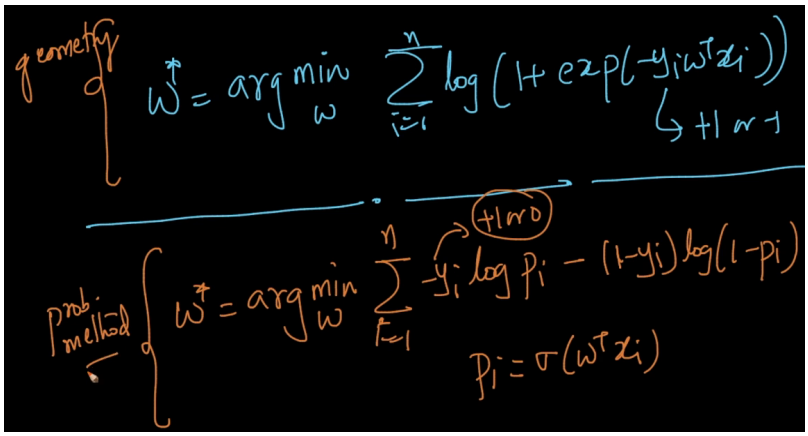
$$f(x) = \frac{1}{1+e^{-(x)}}$$

Monotonic Function-

If x increases then f(x) is also increases.
Eg log(x)



Mathematical formulation of Objective Function



Weight Vector

Optimal w* of above equation is called Weight Vector.

Case 1. Given a feature fi, if the weight vector component corresponding to feature fi is positive, then for any query point xq, if the feature value corresponding to the ith feature increases, its probability of belonging to +ve class increases.

Case 2. Given a feature fi, if the weight vector component corresponding to feature fi is negative, then for any query point xq, if the feature value corresponding to the ith feature increases, its probability of belonging to -ve class increases.

Need of Regularization -

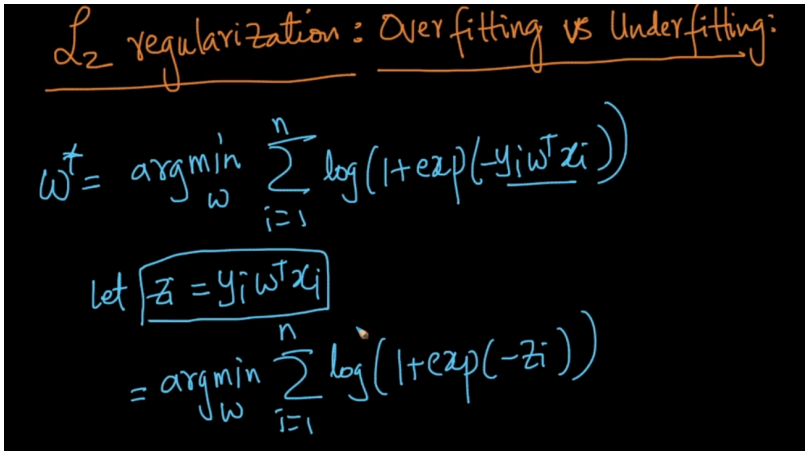
We all know Machine learning is about training a model with relevant data and using the model to predict unknown data. By the word unknown, it means the data which the model has not seen yet. We have trained the model, and are getting good scores while using training data. But during the

process of prediction, we found that the model is underperforming when compared to the training part. Now, this may be a case of **over-fitting** which is causing incorrect prediction by the model. Regularizing the model, in this case, can help resolve the problem.

Regularization -

Regularizations are techniques used to reduce the error by fitting a function appropriately on the given training set and avoid overfitting.

It is a technique which make slight modification to the learning model/algo such that model generalizes better. This in turn improves the model's perfromance on the unseen data as well.



w = *argmin* $\sum_{i=1}^{n}$ *log(1+exp(-zi) >= 0, So minimal value of w* is 0.

Ques- When does this zero occur?

Ans- If zi is positive and tends to + infinity then exp(-zi) tends to become 0.

1. zi = +ve, It means xi is correctly classfied by w
2. zi tends to infinity, then to reach to minima(reach to least value ie 0)

If we pick w such that- As yi and xi are fixed so we can modify only w

- All training point are correctly classfied
- zi tends to +ve infinity. Then such a w is Best "w".

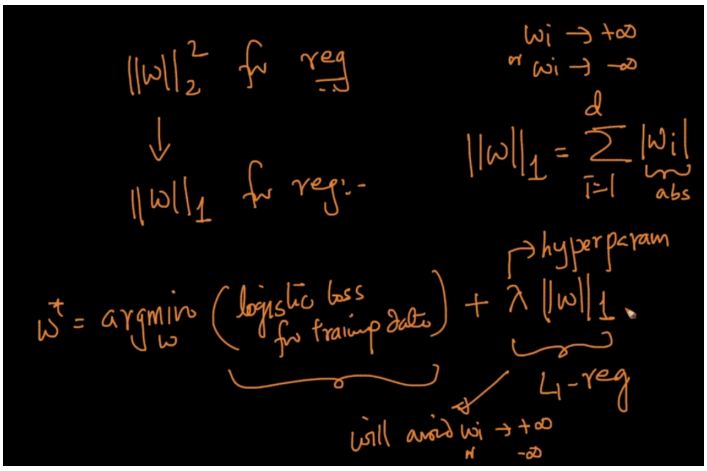
With best w if there are outlier then it cause overfitting. Or wi = +infinity or -infinity To get rid of +infinity and -infinity we use Regularization.

L1 Regularization -

L1 is also called Lasso Regression.

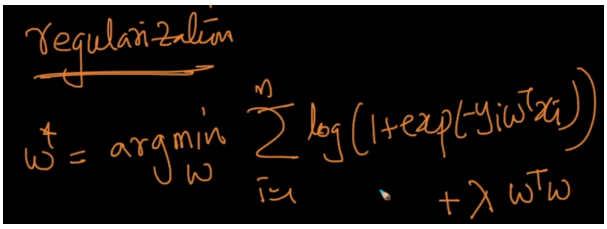
L1 is differ from L2 only in penalizing the high coefficients. It uses |Wj|(modulus)instead of squares of W, as its penalty. In statistics, this is known as the L1 norm.

In the case of the L1, penalty has the effect of forcing some of the coefficient estimates to be exactly equal to zero when the tuning parameter λ is sufficiently large. Therefore, the lasso method also performs variable selection and is said to yield sparse models.

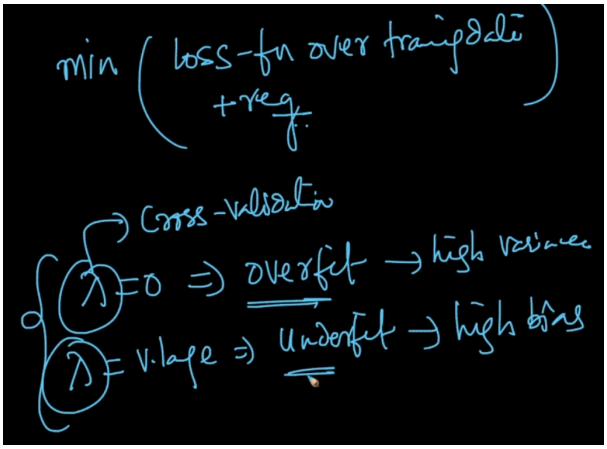


- One benefit we get from L1 is Sparsity.

L2 Regularization -



- First term is called Loss-term.
- Second part of eqn is called Regularization-term
- Second part will ensure that wi dont tends to +infinity and -infinity



Here, λ is the tuning parameter that decides how much we want to penalize the flexibility of our model. The increase in flexibility of a model is represented by increase in its coefficients, and if we want to minimize the above function, then these coefficients need to be small.

L1 Regularization is also called Lasso Regression.
L2 Regularization is also called Ridge Regression.

What does Regularization achieve?

A standard least squares model tends to have some variance in it, i.e. this model won't generalize well for a data set different than its training data.

Regularization, significantly reduces the variance of the model, without substantial increase in its bias. So the tuning parameter λ, used in the regularization techniques described above, controls the impact on bias and variance. As the value of λ rises, it reduces the value of coefficients and thus reducing the variance.

Till a point, this increase in λ is beneficial as it is only reducing the variance(hence avoiding overfitting), without loosing any important properties in the data. But after certain value, the model starts loosing important properties, giving rise to bias in the model and thus underfitting. Therefore, the value of λ should be carefully selected.

Regularization - https://towardsdatascience.com/regularization-in-machine-learning-76441ddcf99a (https://towardsdatascience.com/regularization-in-machine-learning-76441ddcf99a)

Sparsity -

$$w =< w1, w2, w3, \ldots wd >$$

- If w vector is sparse then solution of logistic Regression is said to be Sparse.
- **If we use L1 reg then all less-important feature become zero.**
- **If we use L2 reg then all less-important feature become small but not neccessarily zero.**

$$f1, f2, ..fi...fd$$

- Above are Features, fi is less important feature
- For fi corresponding weight vector is wi.

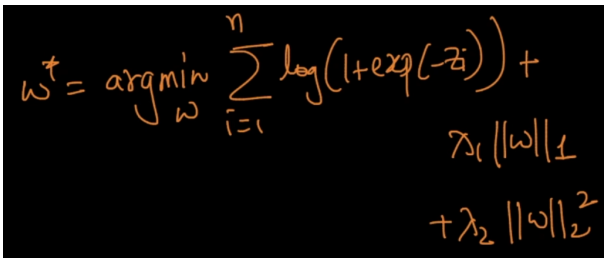
$$w =< w1, w2..wi, \ldots wd >$$

- For L1 reg. wi become zero.
- For L2 reg. wi become small value but not neccsisarily zero.

Linear regression is the standard algorithm for regression that assumes a linear relationship between inputs and the target variable. An extension to linear regression involves adding penalties to the loss function during training that encourage simpler models that have smaller coefficient values. These extensions are referred to as Regularized Linear Regression or Penalized Linear Regression.

Elastic-net -

Elastic net is a popular type of regularized linear regression that combines two popular penalties, specifically the L1 and L2 penalty functions.



Brief Summary of Regularization -

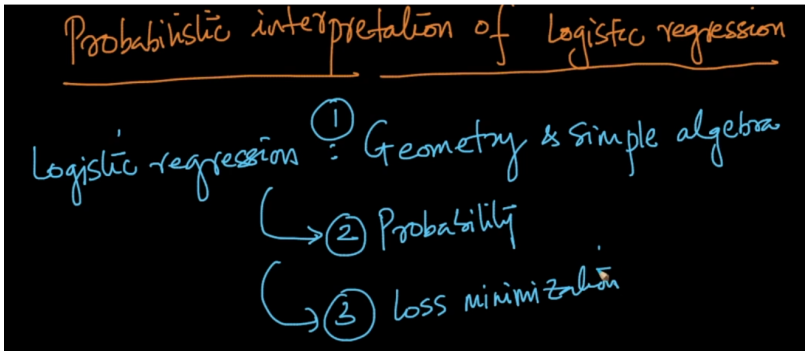
A regularization technique is in simple terms a penalty mechanism which applies shrinkage (driving them closer to zero) of coefficient to build a more robust. Few common ways to regularize models are -

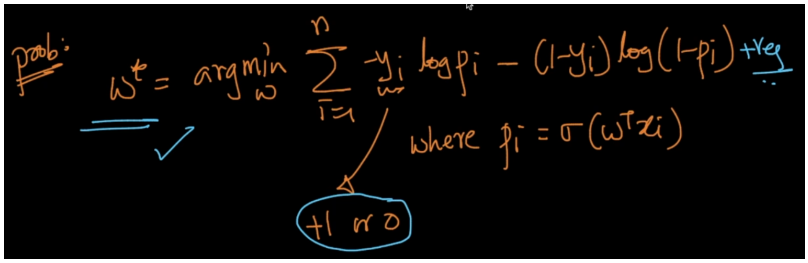
1. L1 Regularization or Lasso Regularization -
This add regularization terms in the model which are function of absolute value of the coefficients of parameters. The coefficient of the paratmeters can be driven to zero as well during the regularization process. Hence this technique can be used for feature selection and generating more parsimonious model

2. L2 Regularization or Ridge Regularization -
This add regularization terms in the model which are function of square of coefficients of parameters. Coefficient of parameters can approach to zero but never become zero.

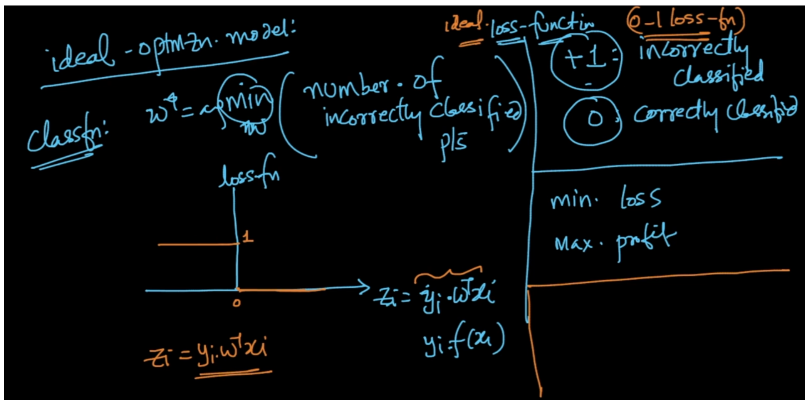
3. Combination of the above two such as Elastic Nets -
This add regularization terms in the model which are combination of both L1 and L2 regularization.

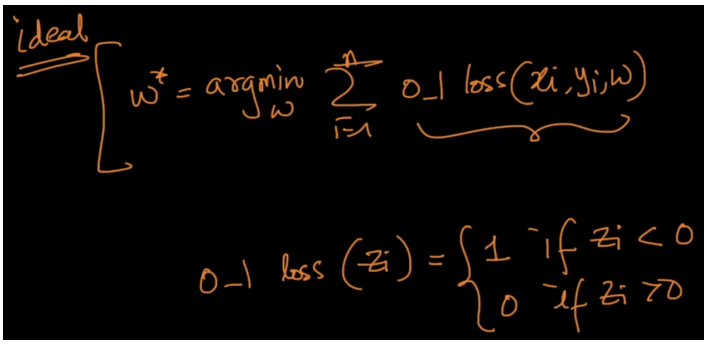
2. Probabilistic Interpretation Gaussian Naive Bayes

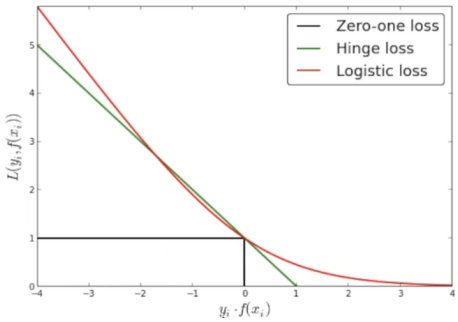




Loss Minimization interpretation of LR

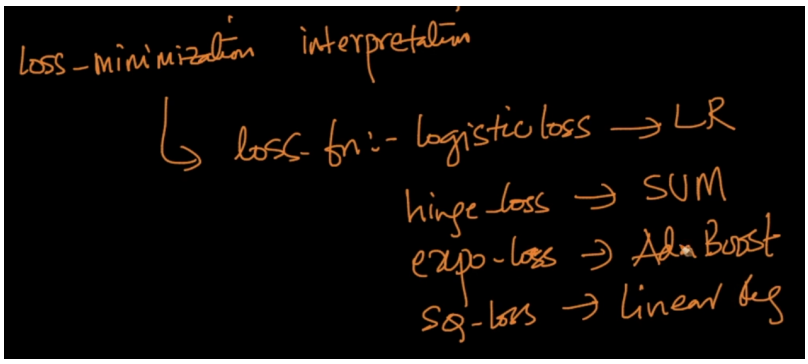






- To minimize the loss function to make the optimize model by using differentiation in calculus.
- f(x) is differentiable when function is continous, if its not continous then its not differentiable.
- Only smooth curves are differentiable.
- Here Hinge-loss is continous but not differentiable.
- As loss func is not continous so its not differentible. So we use Logistic-loss

Loss minimization Interpretation are-

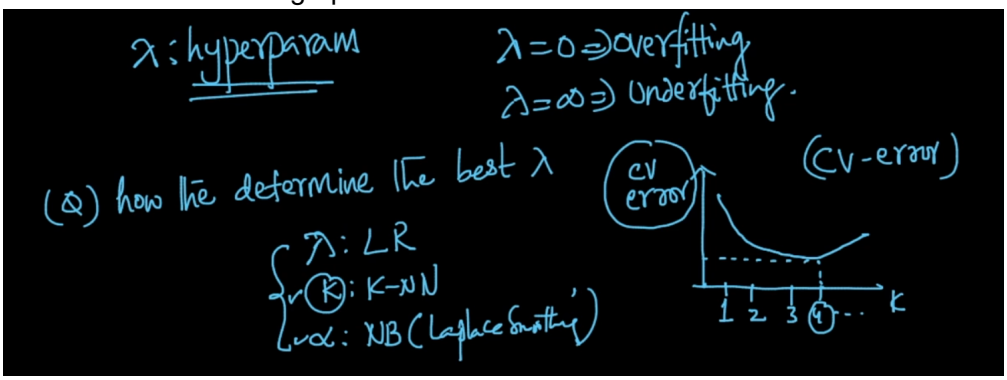


Hyperparameter Tuning -

The hyperparameter is the default parameter that works in all situations. They are termed as an important part of a model. It is not necessary that you can only use the default parameters you can make tweaks if the situation demands. It is important to have three sets in which data is divided like training, testing, validation set whenever you make tweaks in the default parameter to get the necessary accuracy so as to stop data leaks.

- Hyper parameters control the behavior of the algorithm that is used for modeling.
- Hyper parameters are passed in the arguments during the initialization of the algorithm. (Ex. Specifying the criterion for decision tree building)
- If you want to check about the hyperparameters for an algorithm you can make use of the function get_params().

Eg- For k in K-NN we use below graph CV error VS K value.



- K in K-NN is an integer value.
- λ in Logistic Regression is real no.
- To find λ we use Grid Search.

During tuning of the hyper parameters the data should always be divided into three parts that are training, validation, and testing so as to stop data leak.

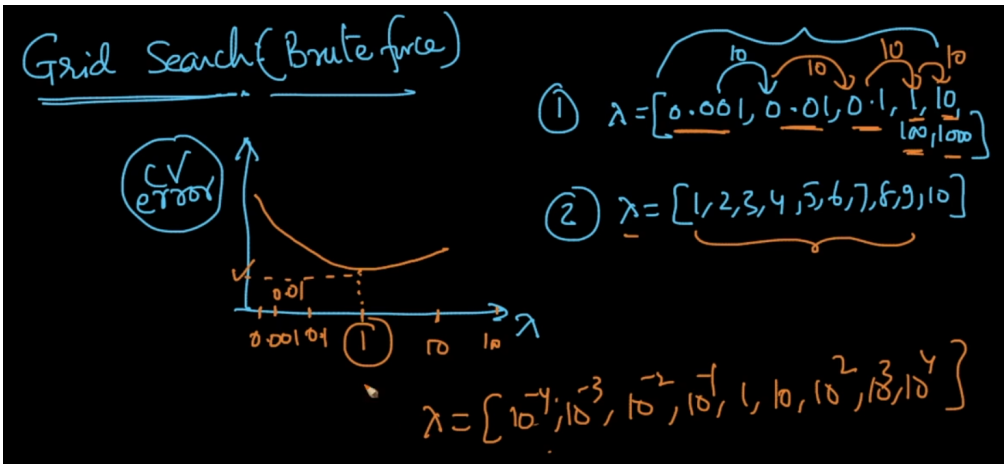
Approch to find Hyper-parameter -

1. **Grid Search CV -** It takes all the parameters combination.
2. **Randomized Seach CV -** It can sample a given number of possibilities with certain distribution from a parameter space.

Grid Search

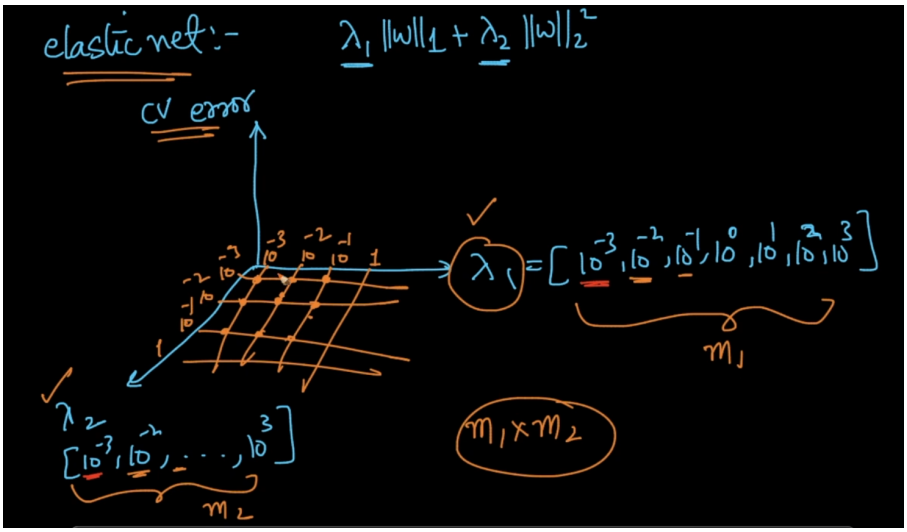
- Grid-searching is the process of scanning the data to configure **optimal hyper-parameter** for a given model.
- It iterates through every parameter combination and stores a model for each combination.
- Every model that is built is validated and ranked.
- The best performing model having best hyperparameters values are taken.

We can take any range of value for λ.



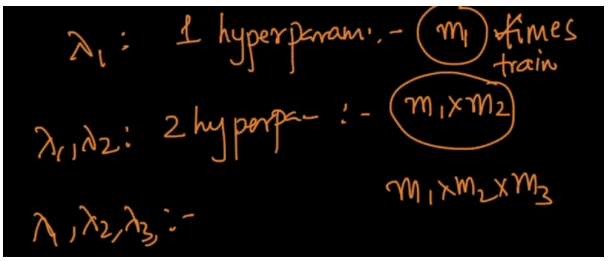
Grid Seach in Elastic-net

- Here we have λ1 and λ2.



In General,

- As hyperparamter increases, the no. of times model needs to be trained increases exponentially.
- Hence Grid Search is not good when are is alot of parameter, So we use **Random Search.**



The **benefit** of grid search is that it is **guaranteed** to find the optimal combination of parameters supplied.
The **drawback** is that it can be very time consuming and computationally expensive.

Random Search-

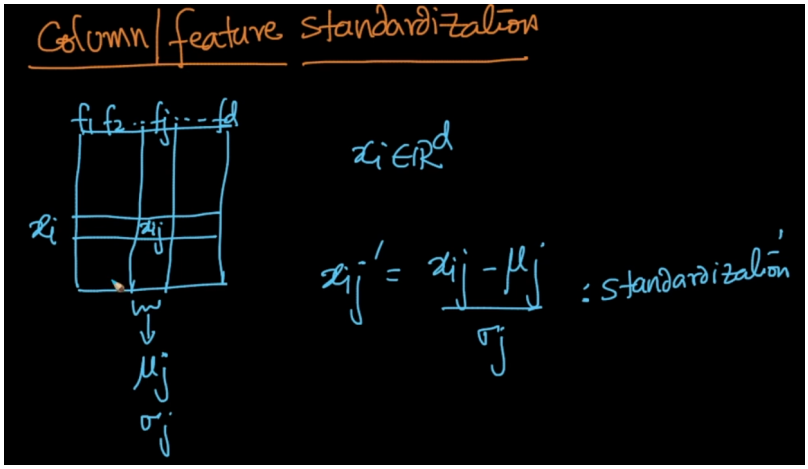
- In Random Search, a random combinations of the hyperparameters are used in a given interval, to find the best solution for the built model.
- It searches the specified subset of hyperparameters randomly instead of exhaustively.
- The major benefit being decreased processing time.
- Its has proven to yield better results comparatively to Grid Search.
- The **drawback** of random search is that it yields **high variance.**

We are **not guaranteed** to find the optimal combination of hyperparameters.

Feature/Column Standardization -

Standardization is another scaling technique where the values are centered around the mean with a unit standard deviation. This means that the mean of the attribute becomes zero and the resultant distribution has a unit standard deviation.

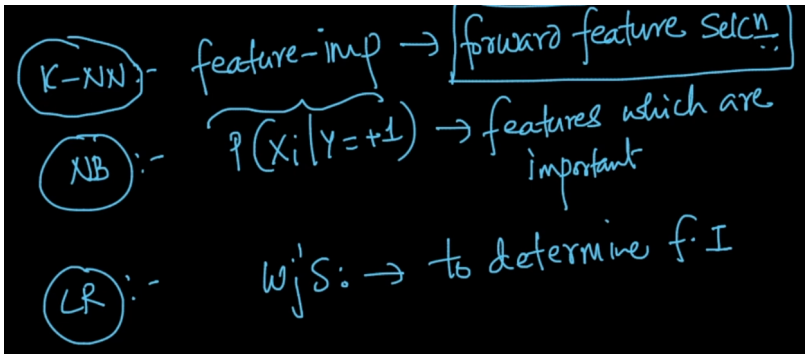
Data Standardization = Feature/Column Standardization = Mean Centering = Scaling all are same.



- Here xij is datapoint,
- μj is Mean of Column
- σj is Standard Deviation of Column

Feature Importance

- Only valid for independent feature.
- If features are independent only then we can use |wj| value of weight vector or value as a Feature importance value.

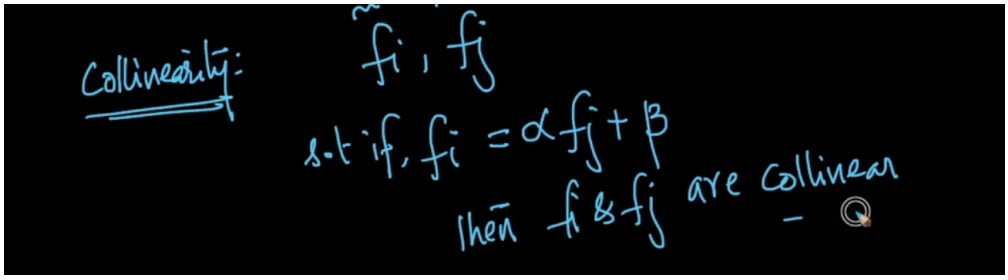


- Here wj is weight corresponding to feature, FI is Feature Importance.

Eg. Given a set of datapoint to distinguish b/w male and female, here we give more weight to hair, hair is considered as one of the feature.

Collinearity or Multicollinearity

- Here when we get another feature by some mathematical operation.
- Below is an eg. of collinear, when we have more than 2 feature which are co-related then its Multicollinearity.



Ques- Why does weight of feature is not useful as Feature Importance if features are Collinear?

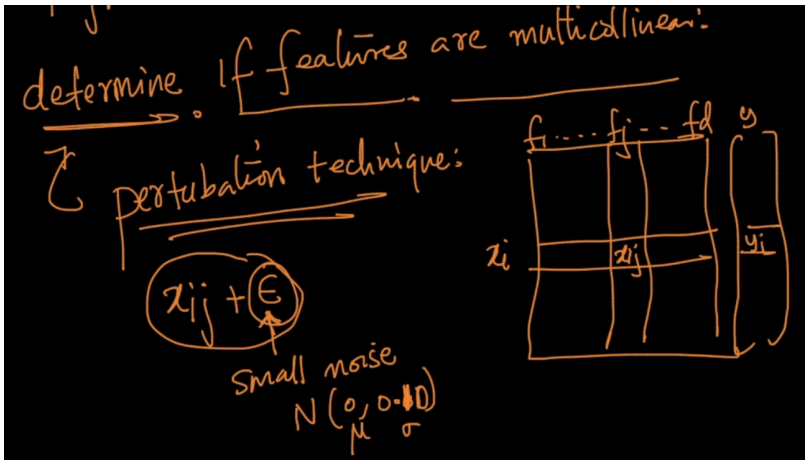
Ans- Different weight of feature will result same classfier.

- If features are collinear then weight vector can change arbitrarily which imply wj or weight vector can not be used as Feature Importance.

Determine Feature are Multicollinear

Pertubation Technique -

- Before Pertubation, we will adding noise we will computer weight vector, w= <w1, w2,...wj...wd>
- Here we add small noise to every xij.
- After pertubation, We will again trained Logistic Regression and compute w̃= <w̃1,w̃2....w̃j...w̃d>.
- If w and w̃ differ drastically then we can conclude that Features are Collinear. It means we can not use |wj| as Feature Importance value.
- If w and w̃ differ slightly then we conclude that |wj| as Feature Importance.



Conclusion - We should always check Collinearity before start using |wj| as Feature Importance value.

- Forward Feature Selection technique can be used in any models

Train & Runtime Space and Time Complexity

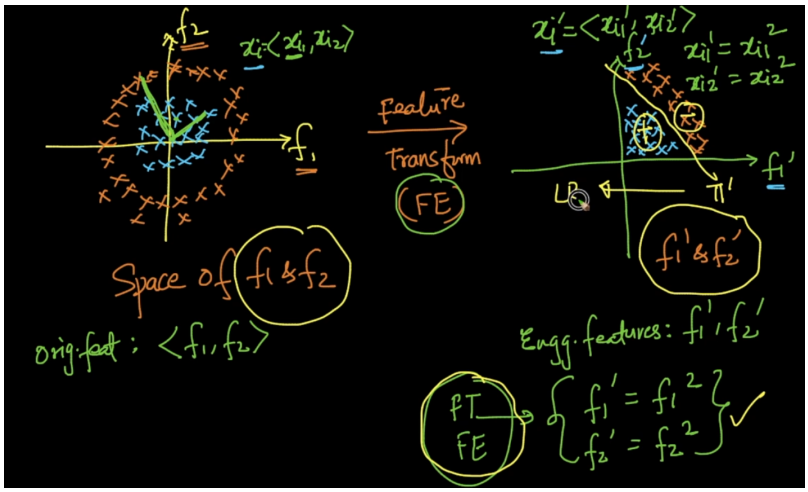
n = No. of points in DTrain, d = Dimension

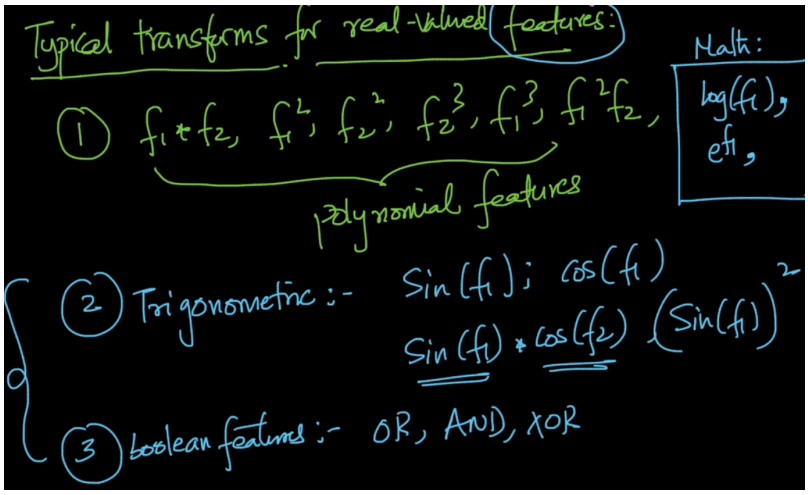
1. Train
   - Time Complexity - O(nd)
2. Runtime
   - Time Complexity - O(d)
   - Space Complexity - O(d)

Non-Linearly separable data & Feature Engineering/ Transform

Ques- For a given datapoints having two concentric circle, Can we distinguish using plane?

Ans- Yes by doing Feature tranform.





Fundamental Techniques of Feature Engineering-

https://towardsdatascience.com/feature-engineering-for-machine-learning-3a5e293a5114 (https://towardsdatascience.com/feature-engineering-for-machine-learning-3a5e293a5114)

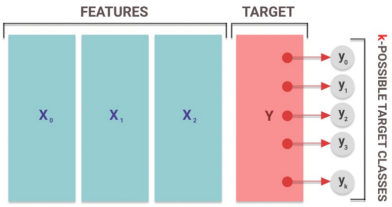
Generalized Linear Model

Log-reg = GNB+ Bernoulli

1. Multinomial LR
2. Linear Regression
3. Poisson Regression

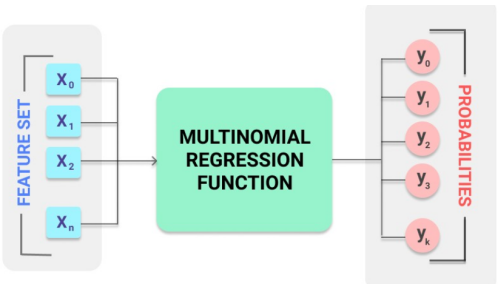
1. Multinomial LR -

MLR can predict one out of k-possible outcomes, where k can be any arbitrary positive integer.



1.1 How does MLR work?

The multinomial regression function is a statistical classification algorithm. What this means is that once we feed the function a set of features, the model performs a series of mathematical operations to normalize the input values into a vector of values that follows a probability distribution.



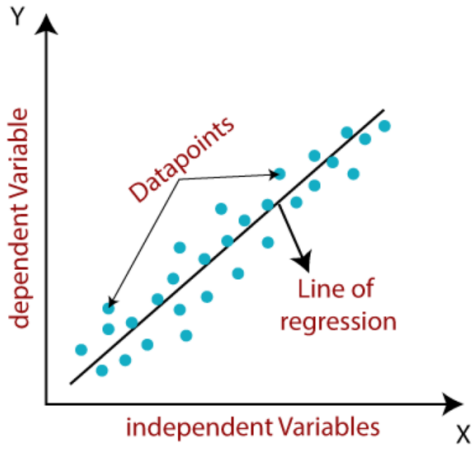
- The input that we give to the model is a feature vector, X, containing features x1, x2, x3…..xn.
- The output we get is a probability vector Y, containing probabilities y1, y2, y3…yk for the k target classes.
- Here, y1 + y2 + y3… + yk = 1, since the total probability of all the possible events in a system is always 1.

Finally, the outcome with the highest probability will be the predicted outcome for the given feature set.

2. Linear Regression -

Linear Regression is a type of regression analysis where the number of independent variables is one and there is a linear relationship between the independent(x) and dependent(y) variable.

Since linear regression shows the linear relationship, which means it finds how the value of the dependent variable is changing according to the value of the independent variable.



3. Poisson Regression -

Poisson regression is used to model count data, assuming that the label has a Poisson distribution. For example, you might use it to predict the number of calls to a customer support center on a particular day.

Refer - https://towardsdatascience.com/generalized-linear-models-8738ae0fb97d (https://towardsdatascience.com/generalized-linear-models-8738ae0fb97d)