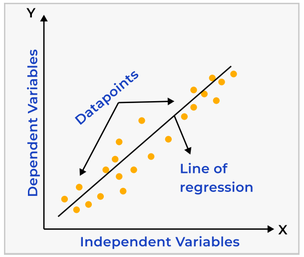
SUPERVISED LEARNING ALGORITHMS

01. Linear Regression



Description

- Simplest algorithm which maps linear relation between input and continuous output by finding the best fine line

$$y = b_0 + b_1x_1 + b_2x_2 + \ldots + b_n * x_n$$

- Assumptions: Linearity ,No multicollinearity, Homoscedasticity, Normality of errors, no autocorrelation

Task	Data Type	Objective Function	Hyperparameter
Regression	Numerical, need to change categorical variables to dummy variables	MSE	Learning rate(η) Regularization(λ)

Algorithm characteristics			
Type	Bias	Variance
Parametric	High	Low

Algorithm characteristics	
Outlier	Multicollinearity
Sensitive, pulls best fit line toward itself	Sensitive

Data Treatment		
Outlier Removal	Missing value treatment:	Feature scaling
Required	Required	Required

Model Interpretability

model coefficients indicate magnitude and direction of each feature effect

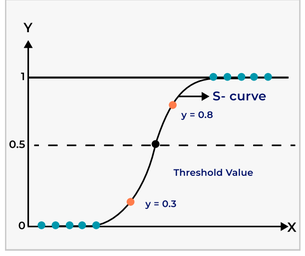
Regularization technique

- L1 (Lasso)
- L2 (ridge)

When to Use

- Model interpretation is required
- less volume of data (with less noise)
- EDA(correlation and scatter plots) indicate linear relationship between input and output

02. Logistic Regression



Description

- Models linear relationship between input and a categorical output (0 or 1) using sigmoid function

$$y = b_0 + b_1x_1 + b_2x_2 + \ldots + b_n * x_n$$

- Predicted class is determined by thresholding the probability, with a commonly used threshold of 0.5.

Task	Data Type	Objective Function	Hyperparameter
Classification	Numerical, need to change categorical variables to dummy variables	Log-Loss or Binary Cross-Entropy (derived from MLE)	Learning rate(η) Regularization(λ)

Algorithm characteristics			
Type	Bias	Variance
Parametric	High	Low

Algorithm characteristics	
Outlier	Multicollinearity
Sensitive, pulls best fit line toward itself	Sensitive

Data Treatment		
Outlier Removal	Missing value treatment:	Feature scaling
Required	Required	Required

Model Interpretability

model coefficients indicate magnitude and direction of each feature effect

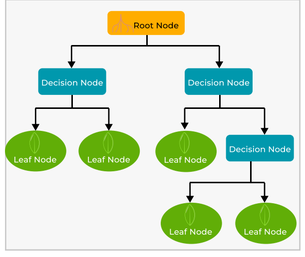
Regularization technique

- L1 , L2 regularization

When to Use

- less data
- model interpretation is required
- classes look geometrically distinguishable

03. Decision Tree



Description

- Uses a set of rules on features to produce predictions.
- Each rule is considered as a node with split being a binary decision which terminates at leaf.
- Criterion like information gain, gini index are used to decide branches while splitting

Task	Data Type	Objective Function	Hyperparameter
Regression Classification	Mixed	• Gini Impurity, Entropy for classification • MSE for Regression	Max Depth, Min Samples Leaf, Min Samples Split

Algorithm characteristics			
Type	Bias	Variance
Non-Parametric	Low	High

Algorithm characteristics	
Outlier	Multicollinearity
No effect due to splitting of data	Robust

Data Treatment		
Outlier Removal	Missing value treatment:	Feature scaling
Not Required	Not Required	Not Required

Model Interpretability

decision tree plots can be used to interpret model

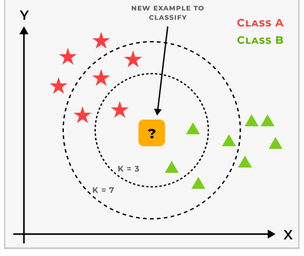
Regularization technique

- Pruning
- optimal tree height

When to Use

- if data has lots of categorical variables
- Large amount of training data points

04. KNN



Description

- Works by finding the K nearest neighbors to the query data point and using their value(class label / regression value) to make a prediction for the target variable.
- A larger K value will result in a smoother decision boundary, while a smaller K value will result in a more complex boundary.

Task	Data Type	Objective Function	Hyperparameter
Regression Classification	Numerical, need to change categorical variables to dummy variables	N/A	distance metric: K (# neighbours), - Euclidian, - Manhattan, or Minkowski

Algorithm characteristics		
Type	Bias	Variance
Non-Parametric	low (if k is small) high (if k is large)	High(k is small) Low (k is large)

Algorithm characteristics	
Outlier	Multicollinearity
Less impact if k is big enough	Sensitive

Data Treatment		
Outlier Removal	Missing value treatment:	Feature scaling
Required	Not Required	Required

Model Interpretability

feature importance can not be interpreted

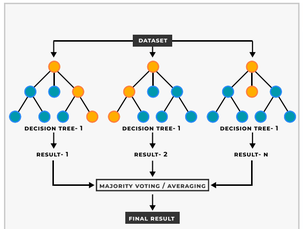
Regularization technique

- k (# neighbours)

When to Use

- small dataset with low dimensions
- where latency is not a concern

05. Random Forest



Description

- Ensemble method that combines the output of multiple decision trees each of which is trained on randomly sampled data with repeatation.

Task

Regression
Classification

Data Type

Mixed

Objective Function

- Gini Impurity, Entropy for classification
- MSE for Regression

Hyperparameter

number of trees, column Sample, row sample size, Depth of base learners

Algorithm Characteristics

Type	Bias	Variance
Parametric	High (bootstrapping and restricted variable participation during split increase bias)	Low

When to Use

- large dataset
- When simple models does not produce desired results

Algorithm Characteristics

Outlier	Multicollinearity
No Impact as it get averaged out due to aggreation	Robust due to column sampling

Data Treatment

Outlier Removal	Missing value treatment:	Feature scaling
Not Required	Not Required	Not Required

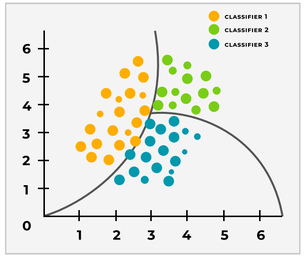
Model Interpretability

Interpretable
Feature importance = weighted information gain of feature across base learners

Regularization technique

base trees
row and column sampling rate

06. Naïve Bayes



Description

- Probabilistic machine learning algorithm based on Bayes' theorem
- Assumes that all the features are independent of each other.

Task

Classification

Data Type

Numerical, need to change categorical variables to dummy variables

Objective Function

N/A

Hyperparameter

Laplace smoothing (α)

Algorithm Characteristics

Type	Bias	Variance
Parametric	High	Low

When to Use

- suitable for solving text classification problems

Algorithm Characteristics

Outlier	Multicollinearity
High impact in Gaussian NB Low impact in Multinomial/ Bernoulli NB	Robust

Data Treatment

Outlier Removal	Missing value treatment:	Feature scaling
Required	Not Required	Not Required

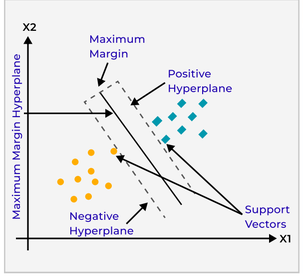
Model Interpretability

Highly interpretable.
Features with high probability are more important

Regularization technique

- Laplace Smoothing

07. SVM



Description

- Works by finding the hyperplane that best separates the data into classes, maximizing the margin between the classes.
- It can even handle non-linear data, by using a technique called kernel trick to map the data into a higher-dimensional space where a linear boundary can be found.

Task

Classification
Regression

Data Type

Numerical, need to change categorical variables to dummy variables

Objective Function

Hinge Loss

Hyperparameter

C gamma kernel

Algorithm Characteristics

Type	Bias	Variance
Parametric	Low	High

When to Use

- less data, and less features - kernel SVM
- less data, more features - linear SVM

Algorithm Characteristics

Outlier	Multicollinearity
Less sensitive, as separating hyperplane is decided by support vector	Sensitive

Data Treatment

Outlier Removal	Missing value treatment:	Feature scaling
Not Required	Required	Required

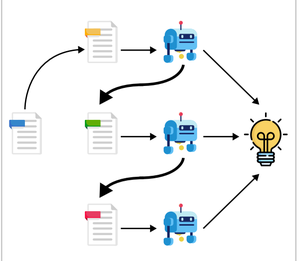
Model Interpretability

Limited to linear SVM
As hyperplane difficult to interpret in high dimensions i.e Kernel SVM.

Regularization technique

- L2 regularization with C parameter

08. Gradient Boosting



Description

- It uses multiple weak models (DT's) in sequential manner where each model tries to predict the error left over by the previous model. GBDT uses gradient descent to optimize the loss function and find the best combination of trees

Task

Classification
Regression

Data Type

Mixed

Objective Function

- Gini Impurity, Entropy for classification
- MSE for Regression

Hyperparameter

- number of trees
- Learning Rate
- Regularization Parameters

Algorithm Characteristics

Type	Bias	Variance
Non-Parametric	Low	High

When to Use

- High dimensional and large datasets

Algorithm Characteristics

Outlier	Multicollinearity
Sensitive, because it builds each tree on previous trees residuals. Outliers will have much larger residuals than non-outliers	Sensitive

Data Treatment

Outlier Removal	Missing value treatment:	Feature scaling
Not Required	Required	Required

Model Interpretability

Interpretable.
Take average of feature importance for all base learners

Regularization technique

base trees
- row and column sampling rate

SUPERVISED LEARNING ALGORITHMS

BAGGING

ensemble methods use predictive power of multiple trees(weak learners) instead of 1.

Follows parallel learning
i.e base learners are formed independently

Random sampling with replacement

Both give final prediction by averaging N learners **But...**

..equal weight is given to all learners

Both provide good scalability **But ...**

..it reduced variance, and helps with overfitting

BOOSTING

But....

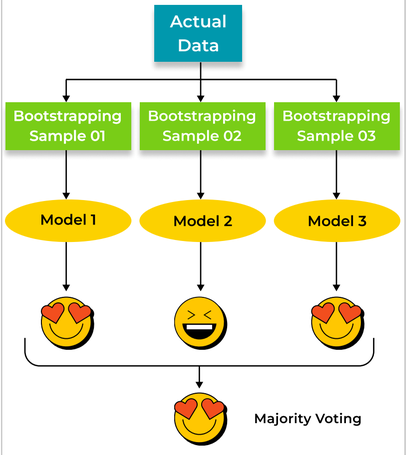
Follows sequential learning
i.e base learners are dependent on previous weak base learner

Random sampling with replacement over weighted data

..more weight is given to learners with better performance (weighted average)

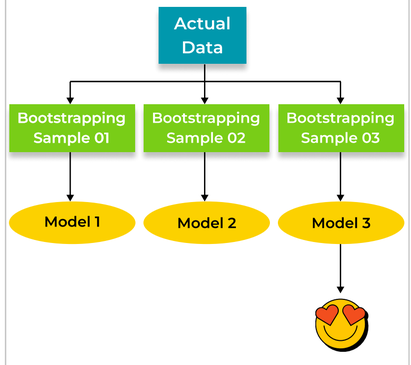
..it reduces bias but more prone to overfitting which can avoided by tuning parameters

BAGGING ENSEMBLE METHOD



BUILD PARALLEL

BOOSTING ENSEMBLE METHOD



BUILD SEQUENTIALLY

REGULARIZATION

L1 REGULARIZATION	L2 REGULARIZATION
How does it penalizes cost function ?	
adds sum of absolute values of of weights	adds sum of squared values of weights
What's the formulation ?	
Cost function = Loss + λΣ|w|	Cost function = Loss + λΣw^2
How does it impact weight coef?	
Produces sparse solution i.e. non important feature weights become 0. (can be used for feature selection)	Produces non sparse solution i.e. reduces the value of non important feature weight but doesn't make them 0
Is it impacted by outliers ?	
Robust to outliers	Impacted by outliers as squared term is involved

REGRESSION METRIC

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|x_i - x|$$

$$MSE = \frac{\sum(y_i - \hat{y}_i)^2}{n}$$

$$RMSE = \sqrt{\frac{1}{n}\Sigma_{i=1}^{n}\left(\frac{d_i - f_i}{\sigma_i}\right)^2}$$

$$R^2 = 1 - \frac{\sum(y_i - \hat{y}_i)^2}{\sum(y_i - \bar{y})^2}$$

$$R^2_{adj} = 1 - (1 - R^2)\frac{n-1}{n-p-1}$$

CLASSIFICATION METRIC

$$Accuracy = \frac{Correct\ Predictions}{Total\ number\ of\ predictions}$$

$$Precision = \frac{TP}{TP + FP} = \frac{True\ Positive}{Predicted\ Positive}$$

$$Recall = \frac{True\ Positive}{True\ Postive\ +\ False\ Negative} = \frac{True\ Positive}{Actual\ Positives}$$

$$Specificity = \frac{True\ Negative}{True\ negative +\ False\ Positive}$$

$$F1 - score = \frac{2}{\frac{1}{Precision} + \frac{1}{Recall}} = \frac{2.Precision.\ Recall}{Precision + Recall}$$

TERMINOLOGIES:

$y_i \rightarrow True\ value$, $\hat{y_i} \rightarrow Predicted\ value$, $n \rightarrow number\ of\ data\ points$

$\bar{y_i} \rightarrow mean\ value\ of\ all\ the\ data\ points$, $k \rightarrow number\ of\ independent\ variables$

PARAMETRIC MODELS

- Fixed number of parameters to build the model.
- Considers strong assumptions about underlying distribution of data
- Requires lesser data

NON -PARAMETRIC MODELS

- Not Fixed i.e flexible
- No or fewer assumptions
- requires more data

BIAS VARIANCE TRADEOFF

$$Error = Bias^2 + Variance + Irreducible\ Error$$

ANALOGY - SHOOTING EXAMPLE

High bias - aiming at wrong place

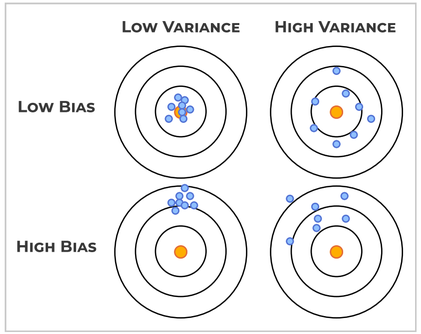
High variance - unsteady aim

Low bias, low variance: Aiming at the target and hitting it with good precision.

Low bias, high variance: Aiming at the target, but not hitting it consistently.

High bias, low variance: Aiming off the target, but being consistent.

High bias, high variance: Aiming off the target and being inconsistent.



High Variance Model ?

Overfitting i.e. High error on test dataset
(performing well on train dataset)

High bias model means?

Underfitting i.e. High error on train as well as test data