

Algorithm	Applications	Pros	Cons
SVM	Medical analytics Text classification Non-linear data	Performs well in high dimensions Best when separable classes Outliers don't impact much Good for binary classification	Slow Poor if overlapping classes Need to select the right hyperparameters Finding right kernel
Naïve Bayes	Text Classification Recommendation Systems!	Performs well in high dimensions Fast- good for real time Scalable Insensitive to irrelevant features Good for multi-class	Training data NEEDS to be representative. Posterior probability gets impacted.
Logistic Regression	Any binary classification	Simple and Effective Doesn't need feature scaling No need to tune hyperparameters	Bad on non-linear data Affected a lot by irrelevant and highly correlated features Not too powerful Need a good representation of data
Random Forest	Recommendation systems Binary classification	Decorrelates trees Reduced error Performs well on imbalanced data Good in high dimensions Can handle missing data well Outliers don't impact too much Overfitting not too much of an issue Good for deducing feature importance!	Features NEED to have predictive power Seems very black box-y : hard to see what different parameters do.
Decision Trees	Feature Importance Fault diagnosis	Doesn't need feature scaling Can handle missing values well Easy to explain Automatic Feature selection!	Tends to get overfitted Sensitive to data Lot of time needed to train
XGBoost	Good for classification	Don't need much feature engineering Fast Outliers don't affect Handles large datasets Good model performance Less prone to overfitting	Visualization is hard Hard to tune - too many hyperparameters

kNN	Good for classification	Easy to implement Doesn't make assumption about data Mutli-class ONE hyperparameter	Slow if large dataset Not good for high dimensional data Scaling MUST be done Doesn't work well on imbalanced data Sensitive to outliers Can't deal with missing values
-----	-------------------------	--	--