Classification

Classification is a supervised learning technique where training data which are accompanied by labels indicating the class of the observations are used for building models and new data is classified based on the model built

It is different from unsupervised technique like Clustering where class label is not known.

In Classification, the dependent variable is categorical whereas in regression analysis the dependent variable is continuous number.

1 Model Evaluation

While there are several classification algorithms, they are generally evaluated using classification accuracy. Those metrics are obtained from confusion matrix. An illustrative example of confusion matrix provided below

Classification Confusion Matrix		
	Predicted Class	
Actual Class	1	0
1	201	85
0	25	2689

201 1's correctly classified as "1"

85 1's incorrectly classified as "0"

25 0's incorrectly classified as "1"

2689 0's correctly classified as "0"

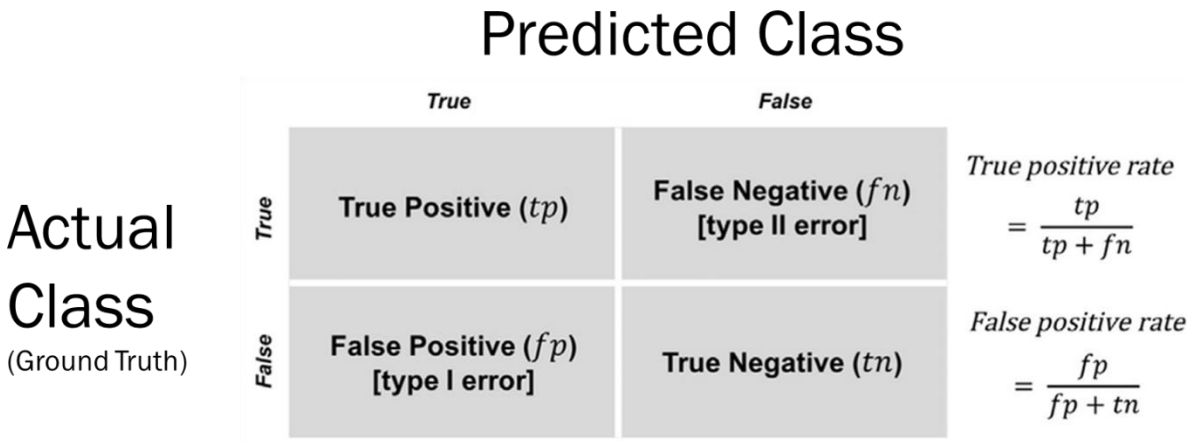
Overall error rate = (25+85)/3000 = 3.67%

Accuracy = 1 – err = (201+2689)/3000 = 96.33%

If multiple classes, error rate is:

(sum of misclassified records)/(total records)

In addition to error and accuracy, other popular evaluation metrics are Sensitivity (True Positive Rate) and Specificity (1 – False Positive Rate)



2 Classification Algorithms

There are several classification algorithms out of which following are very popular

1. Knn
2. Naïve Baye's
3. Decision Tree
4. Bagging (Random Forest)
5. Boosting (Adaboost)
6. Logistic Regression

2.1 KNN – K Nearest Neighbors

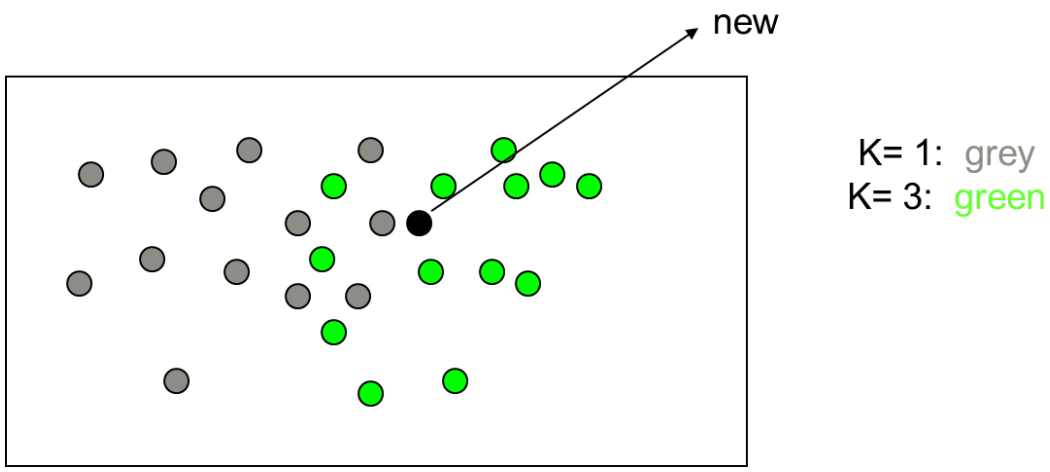
The training examples are vectors in a multidimensional feature space, each with a class label. The training phase of the algorithm consists only of storing the feature vectors and class labels of the training samples.

In the classification phase, following steps are done

– Compute distance (e.g: Euclidean distance) to other training records

– Identify *k* nearest neighbors

– Use class labels of nearest neighbors to determine the class label of unknown record (e.g., by taking majority vote)



2.1.1 Choosing K

If k is too small, classification sensitive to noise points. If k is too large, neighborhood may include points from other classes which could cause confusion. Usual optimal K is identified experimentally.

- Start with k=1 and use a test set to validate the error rate of the classifier.
- Repeat with k=k+2
- Choose the value of k for which the error rate is minimum

K is generally chosen as odd number to avoid ties

2.1.2 Scaling

In classification methods like KNN where distance is used, Independent variables may have to be scaled to prevent distance measures from being dominated by one of the attributes

2.2 Naïve Baye's Classifier

It is a statistical classifier which performs probabilistic prediction based on Baye's theorem.

2.2.1 Bayes Theorem

$$P(H|\mathbf{X}) = \frac{P(\mathbf{X}|H)P(H)}{P(\mathbf{X})} = P(\mathbf{X}|H) \times P(H) / P(\mathbf{X})$$

- Let **X** be a data sample ("evidence"): class label is unknown
- Let H be a hypothesis that X belongs to class C
- Classification is to determine P(H|**X**), (i.e., posteriori probability): the probability that the hypothesis holds given the observed data sample **X**
- P(H) (prior probability): the initial probability. E.g., **X** will buy computer, regardless of age, income, …
- P(**X**): probability that sample data is observed
- P(**X**|H) (likelihood): the probability of observing the sample **X**, given that the hypothesis holds. E.g., Given that **X** will buy computer, the prob. that X is 31..40, medium income

2.2.2 Classifier

Suppose there are *m* classes C_1, C_2, …, C_m, classification is to derive the maximum posteriori, i.e., the maximal P(Ci|X)

$$P(C_i|\mathbf{X}) = \frac{P(\mathbf{X}|C_i)P(C_i)}{P(\mathbf{X})}$$

Since X is multi dimensional, an assumption that the variables are independent is considered, i.e., no dependence relation between attributes

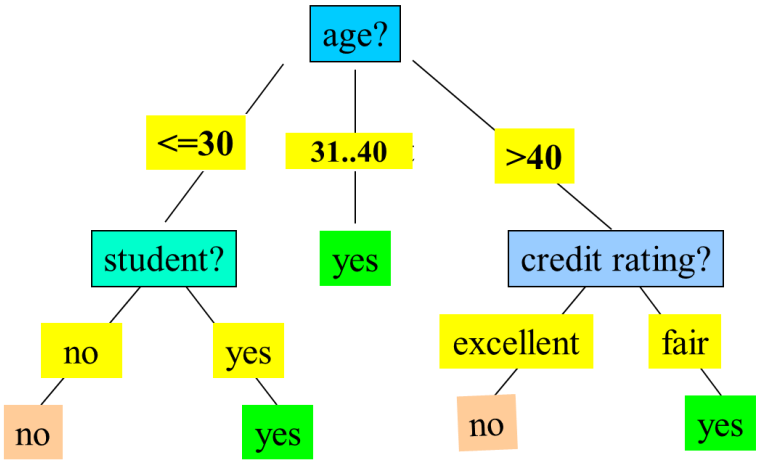
$$P(\mathbf{X}|C_i) = \prod_{k=1}^{n} P(x_k|C_i) = P(x_1|C_i) \times P(x_2|C_i) \times \ldots \times P(x_n|C_i)$$

If the attributes are continuous, Gaussian distribution is assumed for calculating probability (likelihood)

$$g(x, \mu, \sigma) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

2.3 Decision Tree

It is another classification technique where we split the observations into groups based on most significant splitter / differentiator in input variables which maximally differentiates between classes



age	income	student	credit_rating	buys_computer
<=30	high	no	fair	no
<=30	high	no	excellent	no
31…40	high	no	fair	yes
>40	medium	no	fair	yes
>40	low	yes	fair	yes
>40	low	yes	excellent	no
31…40	low	yes	excellent	yes
<=30	medium	no	fair	no
<=30	low	yes	fair	yes
>40	medium	yes	fair	yes
<=30	medium	yes	excellent	yes
31…40	medium	no	excellent	yes
31…40	high	yes	fair	yes
>40	medium	no	excellent	no

2.3.1 Algorithm steps

- Tree is constructed in a top-down recursive divide-and-conquer manner
- At start, all the training examples are at the root
- Observations are partitioned recursively based on selected attributes
- Attributes are selected on the basis of statistical measure (e.g., Information gain, Gini index, Chi-Square)

2.3.2 Regression Tree vs Classification Tree

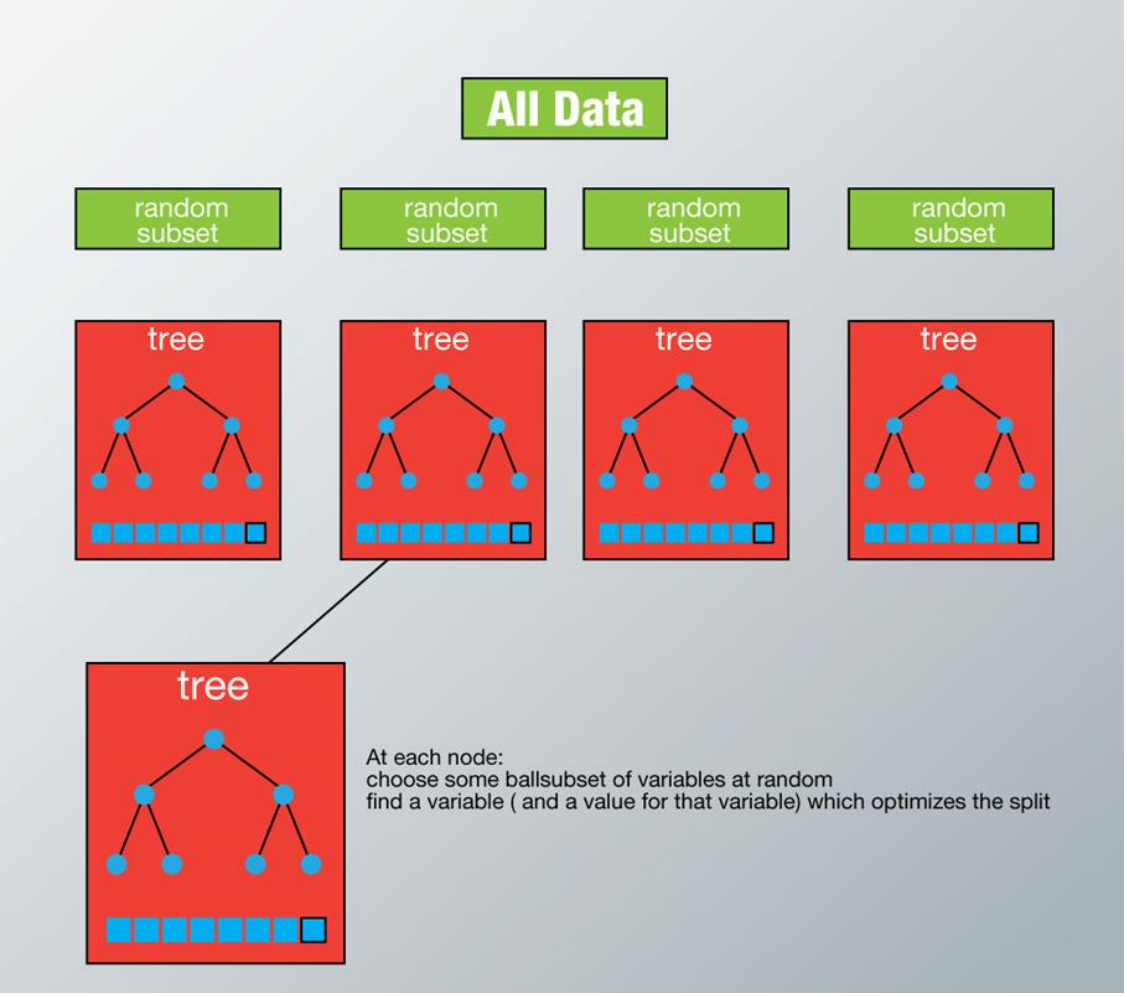
Regression trees are used when dependent variable is continuous. Classification trees are used when dependent variable is categorical. In case of regression tree, the value obtained by terminal nodes in the training data is the mean response of observation falling in that region. Both the trees divide the predictor space (independent variables) into distinct and non-overlapping regions.

2.4 Bagging

Variance is a measurement on how different will the predictions of the model be at the same point if different samples are taken from the same population. Bagging is a technique used to reduce the variance of our predictions by combining the result of multiple classifiers modeled on different sub-samples of the same data set. Random forest is one of the most popular Bagging implementation

2.4.1 Random Forest Steps

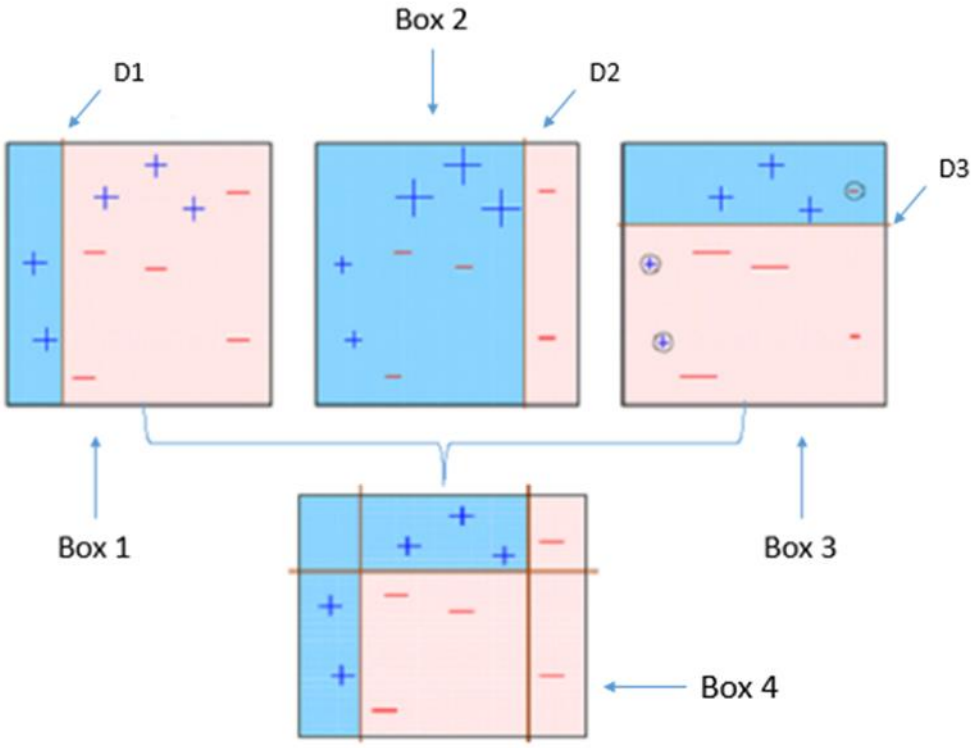
- Assume number of cases in the training set is N. Then, sample of these N cases is taken at random but *with replacement*. This sample will be the training set for growing the tree.
- If there are M input variables, a number m<M is specified such that at each node, m variables are selected at random out of the M. The best split on these m is used to split the node. The value of m is held constant while we grow the forest.
- Predict new data by aggregating the predictions of the ntree trees (i.e., majority votes for classification, average for regression).



2.5 Boosting

Bias is a measurement of how much on an average are the predicted values different from the actual value. Boosting is a classification technique which reduces Bias by converting weak learner to strong learners.

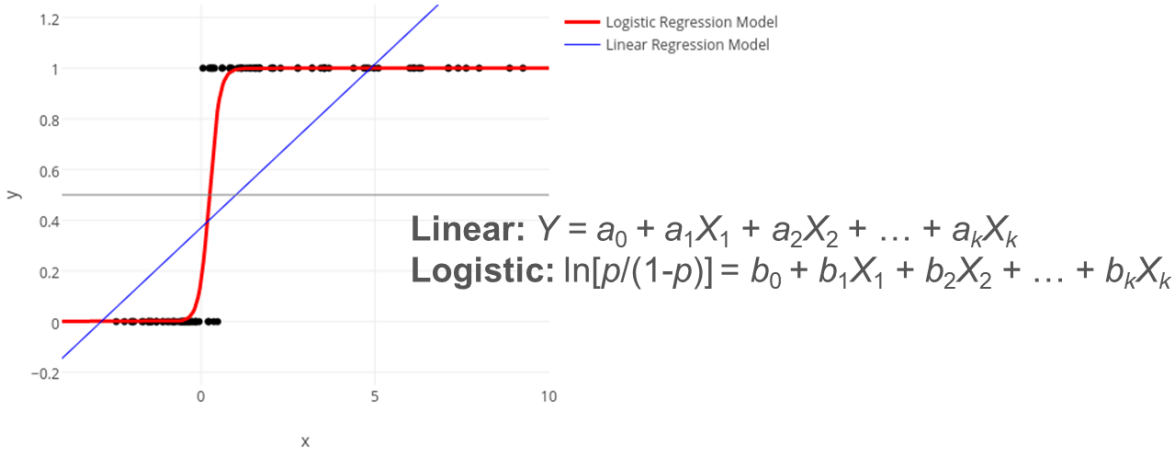
It is an iterative technique which adjust the weight of an observation based on the last classification. If an observation was classified incorrectly, it tries to increase the weight of this observation and vice versa.



Some of the popular Boosting algorithms are AdaBoost (Adaptive Boosting), Gradient Tree Boosting, XGBoost etc.

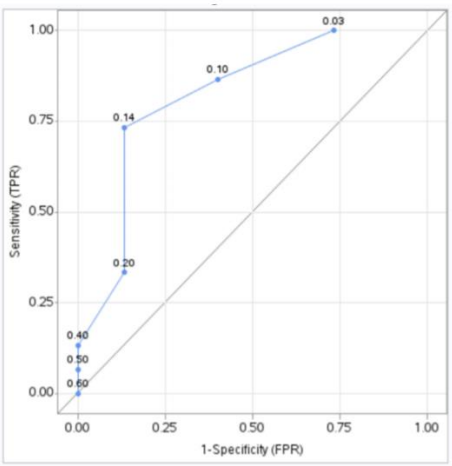
2.6 Logistic Regression

It is a very popular algorithm for 2 class problems. In linear regression, the outcome (dependent variable) is continuous. It can have any one of an infinite number of possible values. In logistic regression, the outcome (dependent variable) has only a limited number of possible values.



2.6.1 ROC Curve

Outcome of Logistic Regression is probability. Cutoff threshold has to be parameterized for classifying between 2 classes. An optimal cutoff is identified which has less FPR and high TPR.



Threshold Cut-off point	Sensitivity	Specificity	1-Specificity (FPR)	Plot Points (X,Y)
≥0.03	1	0.266	0.733	(0.733,1)
≥0.1	0.866	0.6	0.4	(0.4,0.866)
≥0.14	0.733	0.866	0.133	(0.133,0.733)
≥0.2	0.333	0.866	0.133	(0.133,0.333)
≥0.4	0.133	1	0	(0,0.133)
≥0.5	0.066	1	0	(0,0.066)
≥0.6	0	1	0	(0,0)

2.6.2 AUC

Area Under the ROC Curve is used for evaluating the model. AUC ranges between 0.5 and 1 where 0.5 indicates a random model and 1 indicates an ideal model.