**Classification**

According to machine learning, classification is an illustration of supervised learning.

In the Classification problem, we split out our input data into discrete categories. By discrete, we mean separate classes. To predict the category we use classification technique whereas in regression we used to predict a continuous number. From medicine to marketing there is a wide scope of classification applications. A category is a group of people or things sharing particular characteristics.

For examples -

1. Whether food is made from grains or not. Which category does it belong to?

2. To assign a diagnosis to a given patient, based on observed characteristics of a patient.

The classification model includes linear models as well as non-linear ones. Linear models like Logistic Regression, SVM and non-linear like K-NN, Kernel SVM, and Random Forests.

Machine Learning Classification models:

1. Logistic Regression
2. K-Nearest Neighbors (K-NN)
3. Support Vector Machine (SVM)
4. Kernel SVM
5. Naive Bayes
6. Decision Tree Classification
7. Random Forest Classification

**Logistic regression:**

Logistic regression is an analytical analysis which is used to predict dependent variables that takes any one of the categories rather than a continuous outcome.

**Linear equation + sigmoid function ------> Logistic Regression .**

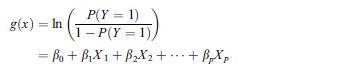
LR is a regression method for predicting a nominal dependent variable which has only two categories. In producing the LR equation, the statistical significance of the variables was determined by the maximum-likelihood ratio (Hosmer & Lemeshow, 2000; Ozdamar, 2004).

To predict the presence or absence of a characteristic or result based on values of a set of predictor variables logistic regression is very helpful. It is alike linear regression model but is suitable for models where the dependent variable is dichotomous.

LR model for p independent variables can be written as



Where P(Y = 1) is probability of presence of CAD, and b0,b1, . . . ,bp are regression coefficients. Within the logistic regression model there is a linear model hidden .In Xi the natural logarithm of the ratio of P(Y = 1) to (1 \_ P(Y = 1)) gives a linear model



The g(x) has many of the useful properties of a linear regression model. The independent variables can be a combination of categorical and continuous variables (Hosmer & Lemeshow, 2000; Ozdamar, 2004).

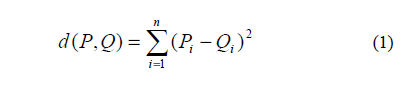
Determining whether there is evidence of interaction and confounder term in the data is the key step in the process of modeling a set of data. The LR model consists of interaction terms and the main effects. The word confounder is used to explain characteristics that are associated with both the primary independent variable and the dependent variable of interest. The relationship between the independent variable and the dependent variable when both associations are present is said to be confounded. LR model is defined to check for the confounder status of a covariate means to compare the approximate coefficient for the independent variable models containing and not containing the covariate.

The covariate is a confounder and should be included in the model is identified Any clinically important change in the estimated coefficient for the independent variable, apart from taking the statistical significance of its estimated coefficient. One of the ways to test for the confounder and interactions in LR is, to begin with, the main effects model, and then use a forward-selection method to search for interaction terms which significantly reduce the likelihood ratio test statistic (Hosmer & Lemeshow, 2000).

**KNN Classifier:**

K-NN comes into a picture where two categories are present in our dataset and a new data point is added in the dataset and so We have to predict the new data point belongs to which category?

In feature space to classify objects based on closest training, the KNN method is used. A type of instance-based learning or lazy learning where the function until classification is only approximated locally and all computation is deferred.

* The simplest forms of all the machine learning algorithms are the k-nearest neighbor algorithm as they justify KNN outcomes easily. To predict labels of any type the KNN method is used.
* · An object is classified by a mass vote of its neighbors, with the object being assigned to the class most frequent amongst its k nearest neighbors where *k*is a positive integer.
* · It is logically tractable.
* · Local information seems very adaptive. KNN algorithms use the nearest data points, therefore it can take full benefit of local information and form highly adaptive, highly nonlinear decision boundaries for each data point.
* · It is effortlessly implemented in parallel. The algorithm checks against the training table for the k nearest neighbor as it is instance based, for each data point to be scored. Since each data point is independent of the others, the execution of score and search can be conducted in parallel.
* The training datasets are described by n-dimensional numeric attributes... The training datasets are stored in an n-dimensional space. The k-nearest neighbor classifier searches the k training samples which are closest to the unknown sample When a test sample (unknown class label) is given. Closeness is defined in terms of Euclidean distance. The Euclidean distance is between two points P(p1,p2, …. Pn) and Q( q1,q2,…. qn) given by equation (1).

Disadvantages of KNN:

* High Computation cost since it needs to calculate the distance of each test instance to all training samples.
* Requires large memory relative to the size of the training set.
* Low precision rate in multidimensional data sets with inappropriate features.
* There is no thumb rule to determine the value of parameter k.
* Delays the process of modeling the training data
* Used in both precision and classification

For better classification, the objective has taken for determining the value of *k*for PIDD

KNN algorithms:

1. Based on distance assignment and classified using k value, The new instance is checked with the already available cases.
2. Least is the distance assigned as more and More the instances are similar and so vice-versa.
3. An instance is assigned to the class Based on the distance, the instance and *k*value
4. KNN classifier is K dependent means it predicts the outcome based on the value of nearest neighbors. Hence the outcome may not be the same for the different values of K.

**Naive Bayes:**

A naive Bayes is used to predict the class membership probability as it is a simple probabilistic based method(Chen, Huang, Tian, & Qu, 2009; Farid & Rahman, 2010). The NB classifier requires the independence in which the effect of an attribute on a given class is independent of those of other attributes called the class conditional.

There are several advantages:

1. Easy to use

2. Probability generation requires only one scan of training data.

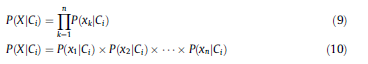
3. Easily handle missing attribute values by simply omitting and comparing the corresponding probabilities.

Given a training dataset, D = {X1,X2, . . . ,Xn}, each data record is represented as, Xi = {x1,x2, . . . ,xn}. D contains the following attributes {A1,A2, . . . ,An} and each attribute Ai contains the following attribute values {Ai1,Ai2, . . . , Aih}. The attribute values can be distinct or continuous. D also contains a set of classes C = {C1,C2, - . . . ,Cm}. Each training instance, X 2 D, has a particular class label Ci. For a test instance, X, the classifier will predict that X belongs to the class with the highest following probability, conditioned on X. That is, the NB classifier predicts that the instance X belongs to the class Ci, if and only if P(CijX) > P(CjjX) for 1 6 j 6 m,j – i. The class Ci for which P(CijX) is maximized is called the Maximum Posteriori Hypothesis.



In Bayes theorem shown in Eq. (8), as P(X) is an invariable for all classes, only P(XjCi)P(Ci) needs to be maximized. If the class preceding probabilities are not known, then it is commonly assumed that the classes are equally possible, that is, P(C1) = P(C2) = \_ \_ \_ = P(Cm), and therefore maximize P(XjCi). Or else, maximize P(XjCi)P(Ci). The class previous probabilities are calculated by P(Ci) = jCi,Dj/jDj, where jCi,Dj is the number of training instances belonging to the class Ci in D. To reduce computation in evaluating P(XjCi), naive assumption of class conditional independence is made. The attributes are conditionally independent with each other, given the class label of the instance.

Thus, Eqs. (9) and (10) are used to produce P(XjCi)

.

n Eq. (9), xk refers to the value of attribute Ak for instance X.

Therefore, training instances help to estimate these probabilities P(x1jCi), P(x2jCi),... ., P(xnjCi) easily. Moreover, the attributes in training datasets can be definite or continuous-valued.

If the attribute value, Ak, is categorical, then P(xkjCi) is the number of instances in the class Ci 2 D with the value xk for Ak, divided by jCi,Dj, i.e., the number of instances belong to the class Ci 2 D.

If Ak is a continuous-valued attribute, then Ak is typically assumed to have a Gaussian distribution with a mean l and standard deviation r, defined respectively by the following two equations:



For all training instances in the class, Ci is the mean and rCi is the standard deviation of the values of the attribute Ak. To estimate P(xkjCi) Now we can bring these two quantities to Eq. (12) together with xk. To predict the class label of instance X, P(XjCi)P(Ci) is evaluated for each class Ci 2 D. The NB classifier predicts that the class label of instance X is the class Ci, if and only if



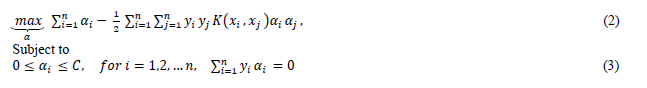
In Eq. (13), 1 6 j 6 m and j – i. That is the predicted class label is the class Ci for which P(XjCi)P(Ci) is the maximum probability. Tables 3 and 4 respectively tabularize the preceding probabilities for each class and conditional probabilities for each attribute value.

**Support Vector Machines**

SVM performs classification tasks by constructing hyperplanes in a multidimensional space that separates cases of unlike class labels. The SVM method 24 is used to maximized the margin between two groups as it provides an optimally separating hyperplane. Support Vectors(SV) are a division of data instances used to explicate the hyperplane. The distance between the nearest support vector and the hyperplane is called margin 25. SVM supports both classification and regression tasks and can handle multiple continuous and definite variables. Two types of SVMs are their (a) Linear SVM which is used to separate the data points using a linear decision boundary and (b) Non-linear SVM which separates the data points using a nonlinear decision boundary 7.

Quadratic programming (QP) package is been used by Traditional SVM training algorithms. Solving a quadratic programming problem is time-consuming and requires a lot of memory as well as in-depth knowledge of mathematical analysis.

Consider a binary classification problem with a dataset (x1, y1),(x2,y2), ..., (xn, yn), where xi is an input vector and yi{-1, +1} is a binary label corresponding to it. The dual form of representing the quadratic programming problem is given below 11.



Sequential Minimal Optimization(SMO) does away out with the need for quadratic programming. working with numerical QP routines is been avoided by the Sequential Minimal Optimization (SMO) algorithm (Platt, 1999a) by logically solving a large number of small optimization sub-problems that involve only two Lagrange multipliers at the time.

For any two multipliers, the constraints are reduced to the following:



By finding a minimum of a one-dimensional quadratic function the problem can be solved logically.

The algorithm works as:

1. Find a Lagrange multiplier that violates the Karush–Kuhn–Tucker (KKT) conditions for the optimization problem.
2. Pick another multiplier and optimize the pair.
3. Steps 1 and 2 are repeated until convergence.

When KKT 26 is satisfied by the Lagrange multipliers the problem of quadratic programming is solved. To increase the rate of convergence the above algorithm guarantees convergence and heuristic measures are used to choose the pair of multipliers.

**Random Forest :**

RF is a supervised learning method, which means that each case or sample is labeled with the outcome (class). RF consists of a collection of k classifiers h1(x), h2(x). . . hk(x),with h(x) being the joint classifier [7,18]. Each classifier hi(x) consists of a decision tree, in which nodes are called attributes. The selection of which attribute is collocated in a node n is performed as follows:

1. a subset of attributes is randomly selected
2. an evaluation measure is applied to the selected attributes according to their capability for providing homogeneity partitions of the samples, and
3. the attribute or node with the highest score is chosen.

In particular, to compute the score we use the change of the Gini impurity3[18], as described in Eq. (1).



where *v*i,jis the j value of the I SNP. Probabilities are predicted according to the instances that reach the n node. Once a node is assigned to an attribute gi, the data is split into as many sets as values the gi attribute has (four). Then, the tree is developed with new nodes in each branch. These are obtained by \_G(gi, n) = − \_Ck ∈ Cp2(Ck) + NVAi \_j=1 p(*v*i,j) \_Ck ∈ Cp2(Ck|*v*i,j) repeating the attribute selection process.

The number of instances that remain in anode decides the stopping condition: if this number is lower than a given threshold, the algorithm stops. Samples used to build each tree are also selected randomly with replacements.

Given a query case q, with a list of SNP-value pairs\_gi, vi, each decision tree provides an outcome, hi(q). The final prediction (class for q) is obtained by using an averaging method that combines the probabilistic prediction of each tree regarding each class. The class with the highest prediction is assigned to q.