KNN is a simple and direct classifying model, it is robust to noisy training data and extremely efficient in handling high dimensional training data. During training, KNN will just accept and persist every training sample in the model. When the model need to predict a new sample, sample distances (such as Euclidean and Manhattan) between the sample and each training sample are calculated. The training samples will then be sorted by ascending order in terms of distance and the result will be the majority group that the first K training samples closest to the testing sample belong to. The description above is the fundamental concept on KNN, one may customize the algorithm by defining the formulation of sample distance or introducing alternative voting scheme to obtain the majority category. Despite the elegance of KNN, its performance is often suboptimal. The main drawback of KNN is that the model cannot reflect any interpretation on data and it is computationally expensive to perform prediction since distances from all training sample need to be calculated.

NB classifier is a simple probabilistic classification model, its prediction result is expressed in probability of each possible outcomes. The model originates from the conditional probability formula Bayes’ rule:

Figure 1: Bayes formula

In the following context, consider as scalar value and as dimensional vectors and can be either 0 or 1. This indicates that the model is trying to solve a binary classification problem with N dimensional feature. Hence the outcome will be in the following when is a new sample to be classified:

Figure 2: NB outcomes for Binary Classification Problem

The probability distribution of the outcome is known as likelihood, the conditional probability of feature given certain outcome is known as prior and is known as posterior which is also the probability of outcome given data. To train the naïve Bayes model, the posterior need to be found by using the training data. This process start with the assumption on likelihood distribution. In this binary classification case, one may normally uses Bernoulli distribution and parameters can be found by maximum likelihood optimization. Next, the prior distribution need to be established and another essential assumption must be made: all features are independent. Thus, the direct evaluation of multivariate distribution is avoided and the prior distribution become product of conditional distribution of each feature given an outcome:

Figure 3: Prior distribution

By using the same approach, the prior distribution of each feature can be found by model assumption and parameterized using maximum likelihood optimization. The last element is a constant as training data is provided so a probability distribution of each feature can be estimated from training data. To apply model to classify a new feature vector , model can just directly evaluate the formula listed in Figure 2 and map to classes with highest probability. The simplicity of training and prediction of the model made itself a very efficient algorithm to work on huge dataset. However, there are obvious limitations that hider its performance. First, the assumption of independence of the prior distribution is often too optimistic for handling realistic data such as biometrics. Secondly, the estimation of likelihood distribution is often inaccurate due to the noise existing in the data. I. Rish performed an empirical study on naïve Bayes classifier [10] and discovered that the model performs well on completely independent or functional independent features.

RF classifier is an ensemble method that composed of decision forest(DT), which is another type of classification model. Ron Kohavi and Ross Quinlan have summarized the most commonly used DT algorithms: C5 and CART [11]. CART stands for Classification and Regression Tree: Binary tree in which each node contains subset of training data. The training process(C5) repeats a binary splitting process on each parent node to generate two subsets of samples base on different splitting criterion until a stopping condition is reached. The splitting criterion is selected so that the information gain (Gini information index) on each split is optimized. After that, a pruning technique is applied to reduce the tree size – cost complexity pruning. When new sample is provided, same series of decision will be made until a leaf node is reached. The outcome will be the category with highest votes from all member in the leaf. There are many alternatives for DT algorithm, such as different split searching methods, criterion function, pruning method and voting schemes. The advantage of DT is that a series of decision for splitting data can be reflected. However, since a DT is solely based on greedy search on splitting, it is very likely to be overfitted, and does not generalized well for prediction on unseen examples. To address this problem, Leo Breiman has provided the major endeavor to formularized RF models. [12] The most popular and fundamental version of RF is a version which establish bootstrapping, bagging and random feature selection. Primarily, RF is just a group of DT predictors with aggregation to conclude prediction outcome by summarizing underlying DTs’ outcomes. Bootstrapping refers to the fact that each DT in the RF is grown by using a randomly drawn subset of training set. During training on each DT, random feature selection causes each node split only consider a randomly picked subset of features. Lastly, the aggregation of all DT results is known as bagging. RF construction has introduced a huge degree of randomness in each DT training and hence prevent overfitting for the training set. Many modification can be applied to RF training, for example, the bagging scheme, bootstrapping method, etc. Apart from preventing overfitting, RF is advantageous in handling high dimensional data in terms of training and prediction method. However, when extrapolation on feature is required for prediction or dealing with temporal correlation, RF is generally suboptimal.

SVM is known as an optimal margin classifier which separate training samples by using linear hyperplane. Consider the following figure,



Figure 4: SVM hyperplane [13]

The “Street” distance is defined as the distance between the closest class member and decision boundary. Formally, let as the normal vector to the decision boundary, as the feature vector and as the binary class label. The decision rule can be formulated as , if the class of sample is 1, vice versa. Given there is available training samples, the optimal is found in training by:

Figure 5: minimax optimal margin problem

To account for scaling problem of , let’s provide another optimization constraint on . Hence, the optimization problem becomes

Figure 6: primal problem of SVM

This optimization problem can be transformed in to duality form by using Langrage Multiplier.

Figure 7: Lagrangian of SVM

For simplicity, we will omit the details of solving optimization problem in Figure 7. In brief, the above expression is solved by using quadratic programming method to obtain and final decision rule become:

Figure 8: Final decision rule of SVM

In the final decision rule, is a kernel method that perform dot product operation on vector and . and are two arbitrarily “support vectors” representing the binary classes. “Support Vectors” are the training samples which has non-zero and hence a linear dependency with the decision boundary. Therefore, they support the decision boundary. The mathematical details above are summarized from Bernhard E. Boser and Isabelle M. Guyon ‘s work [14]. The powerful capability of SVM comes with the availability of kernel. Kernel methods in SVM allow features to be perform either linear or non-linear transformation into higher dimensional space. This enables data which is not linearly separable in input space to be separated in the feature(kernel) space as shown in Figure 9.

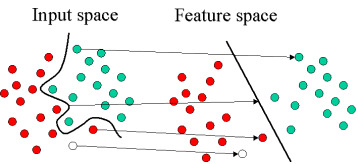


Figure 9: SVM Kernel transform [15]

Commonly kernel methods are linear, polynomial and radial basis function. The advantage of using SVM is that it is prone to overfitting due to maximum margin. On the other hand, the limitation of SVM is the difficulty to derive an effective kernel function and the related optimal hyperparameters.