© INTERNATIONAL JOURNAL FOR RESEARCH PUBLICATION & SEMINAR
ISSN: 2278-6848 | Volume: 13 Issue: 03 | NCASIT- 2022 | April 18th 2022
Paper is available at http://www.jrps.in | Email : info@jrps.in
Refereed & Peer Reviewed

Comparative Analysis of Machine Learning Algorithms : Random Forest algorithm, Naive Bayes Classifier and KNN - A survey

Akshay Gole
Department of Computer Engineering,
St. Vincent Pallotti College of Engineering & Technology,
Nagpur, Maharashtra, India.

Sankalp Singh
Department of Computer Engineering,
St. Vincent Pallotti College of Engineering & Technology,
Nagpur, Maharashtra, India.

Prathmesh Kanherkar
Department of Computer Engineering,
St. Vincent Pallotti College of Engineering & Technology,
Nagpur, Maharashtra, India.

P.R.Abhishek
Department of Computer Engineering,
St. Vincent Pallotti College of Engineering & Technology,
Nagpur, Maharashtra, India.

Prof . Pallavi Wankhede
Assistant Professor
Department of Computer Engineering,
St. Vincent Pallotti College of Engineering & Technology,
Nagpur, Maharashtra, India.

***Abstract*—** Machine learning is a branch of computer science in which a computer predicts the next task to be performed by analysing the data that is provided to it. The computer can access data in the form of digitised training sets or through interaction with the environment. The primary goal of this paper is to provide a general comparison of the Random Forest algorithm, the Naive Bayes Classifier, and the KNN algorithm all aspects. "Random Forest Classifier" is made up of many decision trees. To promote uncorrelated forests, the algorithm leverages randomization to form each individual tree, which then uses the forest's predictive powers to make accurate decisions. The Naive Bayes Classifier is a simple and effective classification method that aids in the development of fast machine learning models capable of making quick predictions. "K-Nearest Neighbour". The algorithm can be used to handle problems involving classification and regression. These algorithms are surveyed on the basis of aim, methodology, advantages and disadvantages

1. INTRODUCTION

Machine learning, in short, is the science of getting computers to act automatically without explicit programming .We've been able to use machine learning for many things over the past decade, from self-driving cars to speech recognition and web search, as well as a vastly better understanding of our genomes. There are a lot of things you probably do every day that utilize machine learning, but you might not even know it.[1] Often, machine learning is classified by how an algorithm improves in its abilities to make predictions. We can categorize learning approaches into four groups: supervised, unsupervised, semi-supervised, reinforcement, and ensemble learning. Based on what type of data scientists want to predict, they choose what algorithm to use.

2. LITRATURE REVIEW

We've covered the relevance of machine learning in this section, as well as the Random Forest method, the Naive Bayes Classifier, and the KNN algorithm.

2.1 Machine learning:

Machine learning, in short, is the science of getting computers to act automatically without explicit programming. We've been able to use machine learning for many things over the past decade, from self-driving cars to speech recognition and web search, as well as a vastly better understanding of our genomes. There are a lot of things you probably do every day that utilize

machine learning, but you might not even know it.[1] Often, machine learning is classified by how an algorithm improves in its abilities to make predictions. We can categorize learning approaches into four groups: supervised, unsupervised, semi-supervised, reinforcement, and ensemble learning. Based on what type of data scientists want to predict, they choose what algorithm to use.

Ensemble learning: To understand the Random Forest machine learning algorithm we need to first understand ensemble learning**.**

[2] The concept of ensemble learning basically refers to a method of making predictions based on the prediction of several different models. Ensemble models are more flexible and less sensitive to data because they combine individual models.

Bagging and **boosting** are most popular ensemble learning methods:

Bagging: A bunch of individual models are trained simultaneously. Data from random subsets is used to train the models

Boosting: Individual models are trained in sequential way. Throughout the learning process, each new model learns from the mistakes of the previous one.[2]

2.2 Random Forest:

Random forests are ensemble models using bagging as the ensemble method and decision trees as the individual model.
As a result of averaging the predictions from the trees, the model performs better than any one decision tree alone.[2]
In a regression problem, the prediction is achieved by averaging the predictions of all the trees. For a classification problem, the prediction is the class label that has the largest majority vote across the trees of the ensemble.

- **Regression**: A prediction is an average of all the predictions in the decision tree.
- **Classification**: The prediction is the class label with the most votes across all decision trees.

A random forest is constructed by putting bootstrapping samples from a training dataset into a large number of decision trees. At each split point in the construction of trees, random forest also selects a subset of input features (columns or variables) from the input, unlike bagging. The process of building a decision tree involves selecting a split point based on the value of each input variable in the data. As the features are reduced to a random subset that can be considered at each split point, the ensemble decision trees become more diverse.

It results in more or less correlated predictions and errors made by each tree in the ensemble. These less correlated trees often perform better than bagged decision trees when their predictions are averaged to make a prediction.[2]

The number of random features to consider at each split point is probably the most important hyperparameter for tuning random forests.
[3] This hyperparameter should be set to 1/3 of the number of input features as a heuristic for regression.

num_features_for_split = total_input_features / 3

[4] This hyperparameter should be set to the square root of the number of input features as a heuristic for classification

N

um_features_for_split = sqrt(total_input_features)

The depth of the decision trees is another important hyperparameter. In addition to more overfitting, deeper trees are less correlated, which may enhance the ensemble performance. 1 to 10 levels of depths may be effective.[2]

As a last step, you can choose how many decision trees will be included in the ensemble. The number is often increased until no further improvements can be observed.

Advantages of Random Forest [5]:

1. In decision trees, it reduces overfitting and improves accuracy.
2. This algorithm is flexible enough to be used for both classification and regression problems.
3. It can be used for both categorical and continuous values.
4. It automates the process of filling in missing values in the data.
5. It uses a rule-based approach, so it does not require data normalization.

Disadvantages of Random Forest [5]:

1. As it builds numerous trees to combine their outputs, it requires a lot of computational power and resources.
2. Additionally, it requires a great deal of time for training since it combines a lot of decision trees.
3. Moreover, because it uses an ensemble of decision trees, it is hard to interpret and does not indicate the significance of each variable.

Applications of Random Forest:

A banking analysis contains a high risk of profit and loss, thus requiring a lot of effort. In the banking industry, customer analysis is one of the most commonly used studies. Random forests are perfect for detecting any fraud transaction or

problems such as calculating the likelihood of a customer defaulting on a loan.[5]

1. It can be used in pharmaceutical industries to assess the potential of a particular medicine or to identify the chemical composition needed for a medicine.

2. In addition, hospitals can use it to identify illnesses suffered by patients, cancer risk in patients, and many other diseases that depend on early diagnosis and research.

2.3 Naïve Bayes

Naïve Bayes is a classification algorithm that works on the concept of Bayes theorem of probability to predict the classes for an unknown dataset. In simpler terms, the Naïve Bayes algorithm classifies each feature of the given dataset independently irrespective of its relation to any of the other features.

The Bayes theorem provides a way for the user to calculate the posterior probability P(c|x) from P(c), P(x), and P(x|c). The equation for the same is

$$P(c|x) = \frac{P(x|c) \times P(c)}{P(x)}$$

Where,

- P(c|x) is the posterior probability of class
- P(c) is the previous probability of class
- P(x|c) is the probability of the predictor of the given class
- P(x) is the previous probability of predictor

Going by the types of Naïve Bayes classifiers there are three types namely:

Multinomial Naïve Bayes

This is generally used when the task at hand is document classification. For example, if we have to classify the document into types like sports magazines or political magazines.

Bernoulli Naïve Bayes

This classifier is similar to the Multinomial Naïve Bayes classifier with the difference being that the Bernoulli classifier only has predictors in boolean variables i.e., they take up values only in the form of Yes or No.

Gaussian Naïve Bayes

In this classifier, the predictors take up continuous values instead of discrete values. Since the values present in the dataset change, the formula for conditional probability changes to,

$$P(x_i|y) = \frac{1}{\sqrt{2\pi\sigma_y^2}} \exp\left(-\frac{(x_i - \mu_y)^2}{2\sigma_y^2}\right)$$

Advantages

- It is much faster and easier to predict classes for a dataset.
- When the assumptions of independence hold, the Naïve Bayes classifier performs well than most of the other existing models
- It performs well with categorical labels than numerical variables

Disadvantages

- The independent assumptions are a big factor in making guesses hence if that doesn't hold then the Naïve Bayes classifiers fail to give the correct output
- If a label is observed in test data but not training data then the model assigns 0 value to it and will be unable to make predictions for it

Applications

Used for Document classification

Used for Email filtering, Spam Filtering

Used for construction of recommendation system which is used for data mining

Used for real time predictions

2.4 K - Nearest Algorithm (KNN Algorithm)

The k-nearest algorithm or the k-nearest neighbors algorithm is a non-parametric supervised learning method which was first developed in the year 1951 by Joseph Hodges and Evelyn Fix, which was later expanded upon by Thomas Cover.

The KNN algorithm is used to solve classification as well as regression problems.

It works by finding the distances between a query and all the examples in the data, selecting the specified number examples (K) closest to the query, then votes for the most frequent label or averages the labels in case of classification or in the case of regression respectively.

In the case of classification and regression, choosing the right K for the data is done by trying several different Ks and then picking the one that works best according to our needs.

Advantages

KNN is widely used because of the vast advantages it offers. It is very simple to implement and understand. The KNN algorithm has no explicit training step and all the work happens during prediction itself. As new data is added to the data-set, the prediction is adjusted without having to retrain a new model as there is no explicit training set for it. Also, since there is only a single hyper parameter, i.e, the value of K, hence hyper parameter tuning becomes pretty easy.

Disadvantages

like all other algorithms, the KNN algorithm isn't perfect either. When there is a high amount of data set to process, the prediction complexity becomes very high. Also for higher dimensional data too, the prediction complexity becomes high. The KNN algorithm is sensitive when features like distance, dimension etc. have different ranges. Also noisy data can result in over-fitting or under-fitting of data.

Application

The KNN algorithm has applications in various fields. Some of the common applications of KNN are as follows:
1. Facial recognition systems.
2. Recommendation Systems.
3. In the agricultural sector to predict various factors.

The KNN algorithm is used in different platforms such as on Netflix or on Amazon where the user or the customer is given recommendation for movies, series, products etc, based on their previous searches or watch history.

Clarity on Classification prediction	Best	Best	Average
Parameters handling for model	Average	Best	Average
Overall accuracy	Best (84.13%)	Worst (80.14%)	Good (83.65%)

3.Comparative Analysis

This section provides a comparison of the above-mentioned algorithms with respect to a few important parameters. In the end, we evaluate the overall accuracy of these algorithms. This analysis is based on the earlier mentioned dataset.

Table 1 An analysis of three widely used supervised classification algorithms.[6]

Parameters for comparison	**Random Forest**	**Naive Bayes**	**k-NN**
Speed of learning	Average	Best	Best
Classification speed	Best	Best	Worst
Performance when value is missing	Average	Best	Worst
Performance with irrelevant features	Average	Good	Good
Noise tolerance	Good	Average	Average
Performance on discrete/binary attributes	Good	Average	Average

The above is the comparison of widely used and most popular supervised classification algorithms. Accuracy is determined by comparing the confusion matrix. As a measure of performance, accuracy is the ratio of correct predictions to all observations. It is the most intuitive measure. The accuracy is compared by applying the algorithms to the dataset. [6]

4. Conclusion

We examined three basic algorithms in depth in this comprehensive survey: the Random Forest method, the Naive Bayes Classifier, and the KNN algorithm. These three algorithms were compared based on a number of parameters. This paper will aid researchers in determining which one of these three algorithms is the best to use in their future research.

REFERENCES

[1https://www.geeksforgeeks.org/machine-learning/

[2] https://machinelearningmastery.com/random-forest-ensemble-in-python/

[3] Page 199, Applied Predictive Modeling, 2013

[4] Page 387, Applied Predictive Modeling, 2013.

[5] https://www.mygreatlearning.com/blog/random-forestalgorithm/#AdvantagesandDisadvantagesofRandomFore st

[6] [6] Sen, Pratap & Hajra, Mahimarnab & Ghosh, Mitadru. (2020). Supervised Classification Algorithms in Machine Learning: A Survey and Review. 10.1007/978-981-13-7403-6_11.