Machine Learning Classification Algorithms

Artificial Intelligence Fundamentals

Ipiña Zarazúa José Alfredo

Ingeniería en Computación, Gen. 2018 Grupo: 281303

UASLP Universidad Autónoma de San Luis Potosí
FACULTAD DE INGENIERÍA

March 20, 2023

Machine Learning Classification Algorithms

Abstract

Machine learning is a branch of artificial intelligence (AI) and computer science which focuses on the use of data and algorithms to imitate the way that humans learn, gradually improving its accuracy.

Machine learning is an important component of the growing field of data science. Through the use of statistical methods, algorithms are trained to make classifications or predictions, and to uncover key insights in data mining projects. These insights subsequently drive decision making within applications and businesses, ideally impacting key growth metrics. Some of those algorithms will be described next.

Naïve Bayes Algorithm

Description

Naive Bayes classifiers are a collection of classification algorithms based on Bayes' Theorem. It is not a single algorithm but a family of algorithms where all of them share a common principle, i.e. every pair of features being classified is independent of each other.

The dataset is divided into two parts, namely, feature matrix and the response vector.

Feature matrix contains all the vectors(rows) of dataset in which each vector consists of the value of dependent features.

Response vector contains the value of class variable(prediction or output) for each row of feature matrix.

The fundamental Naive Bayes assumption is that each feature makes an:

- independent
- equal

contribution to the outcome.

Comparison with other algorithms

Advantages of Using Naive Bayes Classifier

- Simple to Implement. The conditional probabilities are easy to evaluate.
- Very fast – no iterations since the probabilities can be directly computed. So this technique is useful where speed of training is important.
- If the conditional Independence assumption holds, it could give great results.

Disadvantages of Using Naive Bayes Classifier

- Conditional Independence Assumption does not always hold. In most situations, the feature show some form of dependency.
- Zero probability problem : When we encounter words in the test data for a particular class that are not present in the training data, we might end up with zero class probabilities.

What kind of problems can the algorithm solve?

It has been successfully used for many purposes, but it works particularly well with natural language processing (NLP) problems. Also can be used in sports to predict results and helping to make decisions and choose the best strategies.

Applications of Naive Bayes

Text Classification. Most of the time, Naive Bayes finds uses in-text classification due to its assumption of independence and high performance in solving multi-class problems. It enjoys a high rate of success than other algorithms due to its speed and efficiency.

Sentiment Analysis. One of the most prominent areas of machine learning is sentiment analysis, and this algorithm is quite useful there as well. Sentiment analysis focuses on identifying whether the customers think positively or negatively about a certain topic (product or service).

Recommender Systems. With the help of Collaborative Filtering, Naive Bayes Classifier builds a powerful recommender system to predict if a user would like a particular product (or resource) or not. Amazon, Netflix, and Flipkart are prominent companies that use recommender systems to suggest products to their customers.

When you should use this algorithm?

- Naive Bayes is suitable for solving multi-class prediction problems.
- If its assumption of the independence of features holds true, it can perform better than other models and requires much less training data.
- Naive Bayes is better suited for categorical input variables than numerical variables.

Support Vector Machines

Description

Support Vector Machine or SVM is one of the most popular Supervised Learning algorithms, which is used for Classification as well as Regression problems. However, primarily, it is used for Classification problems in Machine Learning.

The goal of the SVM algorithm is to create the best line or decision boundary that can segregate n-dimensional space into classes so that we can easily put the new data point in the correct category in the future. This best decision boundary is called a hyperplane.

SVM chooses the extreme points/vectors that help in creating the hyperplane. These extreme cases are called as support vectors, and hence algorithm is termed as Support Vector Machine.

Comparison with other algorithms

Advantages of SVM

- Support vector machine is very effective even with high dimensional data.
- When you have a data set where number of features is more than the number of rows of data, SVM can perform in that case as well.
- When classes in the data are points are well separated SVM works really well.

- SVM can be used for both regression and classification problem.
- SVM can work well with image data as well.

Disadvantages of SVM

- When classes in the data are points are not well separated, which means overlapping classes are there, SVM does not perform well.
- We need to choose an optimal kernel for SVM and this task is difficult.
- SVM on large data set comparatively takes more time to train.
- SVM or Support vector machine is not a probabilistic model so we can not explanation the classification in terms of probability.
- It is difficult to understand and interpret the SVM model compared to Decision tree as SVM is more complex.

What kind of problems can the algorithm solve?
SVM can be of two types: Linear SVM and Non-linear SVM
Linear SVM is used for linearly separable data, which means if a dataset can be classified into two classes by using a single straight line, then such data is termed as linearly separable data, and classifier is used called as Linear SVM classifier.
Non-Linear SVM is used for non-linearly separated data, which means if a dataset cannot be classified by using a straight line, then such data is termed as non-linear data and classifier used is called as Non-linear SVM classifier.
Applications of SVM
Support vector machines are mainly supervised learning algorithms. And they are the finest algorithms for classifying unseen data. Hence they can be used in a wide variety of applications, for example:

- Image-based analysis and classification tasks
- Geo-spatial data-based applications
- Text-based applications
- Computational biology
- Security-based applications
- Chaotic systems control

When you should use this algorithm?
One of the things about support vector machines is that they are more flexible for new data. This makes them easier to use in the applications where we need more flexibility in the training and testing data. Due to the large margin that it likes to generate, we can fit in more data and classify it perfectly.

K-Nearest Neighbors

Description

The k-nearest neighbors algorithm, also known as KNN or k-NN, is a non-parametric, supervised learning classifier, which uses proximity to make classifications or predictions about the grouping of an individual data point. While it can be used for either regression or classification problems, it is typically used as a classification algorithm, working off the assumption that similar points can be found near one another.

The KNN Algorithm

a) Load the data

b) Initialize K to your chosen number of neighbors

c) For each example in the data

    - Calculate the distance between the query example and the current example from the data.
    - Add the distance and the index of the example to an ordered collection

d) Sort the ordered collection of distances and indices from smallest to largest (in ascending order) by the distances

e) Pick the first K entries from the sorted collection

f) Get the labels of the selected K entries

g) If regression, return the mean of the K labels

h) If classification, return the mode of the K labels

It's also worth noting that the KNN algorithm is also part of a family of "lazy learning" models, meaning that it only stores a training dataset versus undergoing a training stage. This also means that all the computation occurs when a classification or prediction is being made. Since it heavily relies on memory to store all its training data, it is also referred to as an instance-based or memory-based learning method.

The goal of the k-nearest neighbor algorithm is to identify the nearest neighbors of a given query point, so that we can assign a class label to that point.

Comparison with other algorithms

Advantages

- Easy to implement: Given the algorithm's simplicity and accuracy, it is one of the first classifiers that a new data scientist will learn.
- Adapts easily: As new training samples are added, the algorithm adjusts to account for any new data since all training data is stored into memory.
- Few hyperparameters: KNN only requires a k value and a distance metric, which is low when compared to other machine learning algorithms.

Disadvantages

- Does not scale well: Since KNN is a lazy algorithm, it takes up more memory and data storage compared to other classifiers. This can be costly from both a time and money perspective. More memory and storage will drive up business expenses and more data can take longer to compute. While different data structures, such as Ball-Tree, have been created to address the computational inefficiencies, a different classifier may be ideal depending on the business problem.
- Curse of dimensionality: The KNN algorithm tends to fall victim to the curse of dimensionality, which means that it doesn't perform well with high-dimensional data inputs. This is sometimes also referred to as the peaking phenomenon (PDF, 340 MB) (link resides outside of ibm.com), where after the algorithm attains the optimal number of features, additional features increases the amount of classification errors, especially when the sample size is smaller.
- Prone to overfitting: Due to the "curse of dimensionality", KNN is also more prone to overfitting. While feature selection and dimensionality reduction techniques are leveraged to prevent this from occurring, the value of k can also impact the model's behavior. Lower values of k can overfit the data, whereas higher values of k tend to "smooth out" the prediction values since it is averaging the values over a greater area, or neighborhood. However, if the value of k is too high, then it can underfit the data.

What kind of problems can the algorithm solve?
k-NN can be used for regression that is, to predict a real-valued property of an unknown item, such as the selling price of a home. With k-NN regression, we would take the selling prices of the k nearest homes, average them, and use that as the predicted selling price.
Applications of KNN Algorithm
The k-NN algorithm has been utilized within a variety of applications, largely within classification. Some of these use cases include:

- Data preprocessing: Datasets frequently have missing values, but the KNN algorithm can estimate for those values in a process known as missing data imputation.

- Recommendation Engines: Using clickstream data from websites, the KNN algorithm has been used to provide automatic recommendations to users on additional content. This research (link resides outside of ibm.com) shows that the a user is assigned to a particular group, and based on that group's user behavior, they are given a recommendation. However, given the scaling issues with KNN, this approach may not be optimal for larger datasets.

- Finance: It has also been used in a variety of finance and economic use cases. For example, one paper (PDF, 391 KB) (link resides outside of ibm.com) shows how using KNN on credit data can help banks assess risk of a loan to an organization or individual. It is used to determine the credit-worthiness of a loan applicant. Another journal (PDF, 447 KB)(link resides outside of ibm.com) highlights its use in stock market forecasting, currency exchange rates, trading futures, and money laundering analyses.

- Healthcare: KNN has also had application within the healthcare industry, making predictions on the risk of heart attacks and prostate cancer. The algorithm works by calculating the most likely gene expressions.

- Pattern Recognition: KNN has also assisted in identifying patterns, such as in text and digit classification (link resides outside of ibm.com). This has been particularly helpful in identifying handwritten numbers that you might find on forms or mailing envelopes.

When you should use this algorithm?

The k-nearest neighbors (KNN) algorithm is a simple, easy-to-implement supervised machine learning algorithm that can be used to solve both classification and regression problems.
KNN's main disadvantage of becoming significantly slower as the volume of data increases makes it an impractical choice in environments where predictions need to be made rapidly. Moreover, there are faster algorithms that can produce more accurate classification and regression results.
However, provided you have sufficient computing resources to speedily handle the data you are using to make predictions, KNN can still be useful in solving problems that have solutions that depend on identifying similar objects.

Conclusions

Machine learning algorithms can be used in a lot of applications, but which algorithm choose depends on dataset you have and also what application are you going to implement, some algorithms will work better than others. Computation and memory resources are important practical considerations, for example SVM only needs a small subset of training points (the support vectors) to define the classification rule, making it often more memory efficient and less computationally demanding when inferring the class of a new observation. In contrast, kNN typically requires higher computation and memory resources because it needs to use all input variables and training samples for each new observation to be classified.

References

Education, I. C. (2022, July 6). Machine Learning. https://www.ibm.com/cloud/learn/machine-learning

Support Vector Machine (SVM) Algorithm - Javatpoint. (n.d.). https://www.javatpoint.com/machine-learning-support-vector-machine-algorithm

Parikh, D. (2020, May 7). Applications of Support Vector Machines (SVM). OpenGenus IQ: Computing Expertise & Legacy. https://iq.opengenus.org/applications-of-svm/

What is the k-nearest neighbors algorithm? | IBM. (n.d.). https://www.ibm.com/topics/knn#:~:text=The%20k%2Dnearest%20neighbors%20algorithm%2C%20also%20known%20as%20KNN%20or,of%20an%20individual%20data%20point.

Harrison, O. (2019, July 14). Machine Learning Basics with the K-Nearest Neighbors Algorithm. https://towardsdatascience.com/machine-learning-basics-with-the-k-nearest-neighbors-algorithm-6a6e71d01761

MLNerds (M.), (2021, August 9) Naive Bayes Classifier : Advantages and Disadvantages https://machinelearninginterview.com/topics/machine-learning/naive-bayes-classifier-advantages-and-disadvantages/

BotBark, (2019, December 19)Top 5 Advantages and Disadvantages of Support Vector Machine Algorithm https://botbark.com/2019/12/19/top-5-advantages-and-disadvantages-of-support-vector-machine-algorithm/