Machine Learning Models
Generative Vs Discriminative

Mason Heaman
Western Washington University
516 High St, Bellingham, WA 98225
mcheaman@yahoo.com

Machine learning classifiers have the primary goal of taking input x and predicting the corresponding output y. However, the membership of a certain model to either the generative or discriminative class is a necessary distinction that can provide a wealth of information on the model's method of prediction. At a fundamental level, discriminative models will model the decision boundary between classes, while generative models will model the actual distribution of each class. In this paper, generative and discriminative models will be compared, contrasted, and explained in detail. Examples of models will be provided to illustrate some of benefits, and shortcomings, seen from both categories of algorithms.

Introduction

A machine learning model is trained on a set of data in order to recognize an overall trend between certain features. Once a model is trained, it can be used to make a prediction on input data that it has never seen before. This ability to recognize patterns in training and apply them to new input is why it is said that a model *learns.* The more data available to a model, the better that it will be able to accurately predict unseen inputs. Therefore, it is no shock that the rise in Machine Learning is partnered with an overwhelming rise in global data generation. In 2020, there was an estimated 59 zettabytes (ZB) of data in the global data sphere, where a single ZB is one trillion gigabytes. This number is predicted to nearly triple to an astounding 175 ZB by 2025 [1]. Machine learning models are essential to fields such as Self-driving cars, Medical Diagnosis, Stock Market trading, and so much more [2]. Given the revolutionary and high-stake applications of Machine Learning, it is essential to discover the model that will perform best for a given application.

Key Differentials

A discriminative model will learn the conditional probability distribution P(Y | X) while a generative model will learn the joint probability distribution P(X,Y). In other words, a discriminative model will model the decision boundary between classes, and a generative model will model the actual distribution of the classes. This can be visualized using the simple binary classification example in **Figure 1.** It is important to note that the decision boundary produced by a discriminative model does not need to be linear. The different methods of classification described above logically lead to another distinction between the two categories: complexity. Generative models tend to take on a much more complex task while discriminative models tend to be simpler.

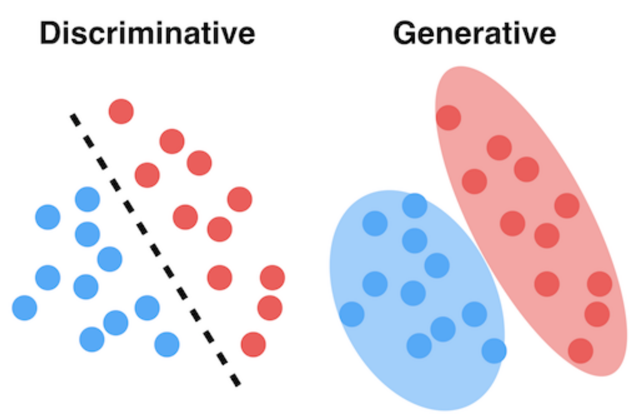

Figure 1: Visual Representation of Discriminative and Generative classifications

Discriminative Pros and Cons

If one's main goal follows the form of a conditional prediction, that is predicting label Y given data X, a discriminative model will generally provide a higher accuracy. This is because discriminative models create decision boundaries by optimizing this form of prediction. Given a large enough dataset, they will be able to effectively generalize to future data sets and outperform generative models of equal complexity. Discriminative models also take on a less complex task than generative models, allowing them to operate faster, use less space, and undergo less computation.

The relative simplicity of discriminative models may also result in disadvantages. In smaller datasets, discriminative models are prone to overfitting and will therefore not generalize well to future data. These models do not provide information on the interactions of model parameters. Therefore, strictly discriminative models are not applicable to tasks that rely on the joint probability distribution of the dataset.

Generative Pros and Cons

As mentioned, generative models tend to take on a much more complex task than that of discriminative models. The richness of these models brings forth advantages and drawbacks. The key advantage of generative models is that they allow one to directly model P(X,Y). This is essential for tasks such as anomaly detection and clustering. In smaller datasets, generative models will generalize better to future data due to a lower likelihood of overfitting. As a generative model learns the distribution among target classes, once trained it can be used to generate synthetic datasets inspired by the original dataset's joint probability distribution. Additionally, generative models allow one to make explicit claims about how the dataset is created.

However, the process in which strictly generative models calculate the posterior probability, the probability of target Y given input X, results in lower performance in conditional prediction tasks. This is because generative models calculate the posterior probability indirectly. Therefore, in a large enough dataset with a conditional prediction goal, generative models will tend to underperform discriminative models of equal complexity[3].

Support Vector Machines

A Support Vector Machine (SVM) is a supervised, discriminative, machine learning algorithm. It can be used for both classification and regression tasks, but they are more commonly used in classification. The fundamental goal of an SVM is to find a hyperplane that will best divide a dataset into classes. Given a binary classification task such as that in Figure 2, the farther that a certain data point is from the hyperplane, the more confident the model will be that the point has been correctly assigned to the respective class. Support vectors are defined as the data points with the smallest distance to a hyperplane, (seen connected to the hyperplane in Figure 2). The distance between a hyperplane and the support vectors of each class is defined as the margin. To find the optimal hyperplane to divide classes, SVM has the goal of finding a hyperplane that maximizes margin [4]. This will result in a model that is more confident in classifying new data points correctly.

As stated in the Discriminative Pros, discriminative models are often favored in classification tasks due to the ability to model the conditional probability distribution directly. SVM follows this trend, and is popular for classification tasks due to these characteristics. However, SVM's goal of creating a hyperplane between sets is not effective in noisy data sets that produce overlapping classes.

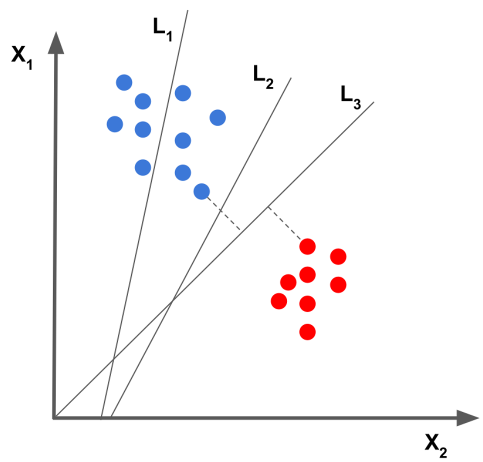

Figure 2: Binary Classification task with SVM

Naïve Bayes

Naïve Bayes is a supervised, generative, machine learning algorithm typically used for classification problems. The fundamental goal of this algorithm is to find the probability that input X will be target Y. We then want to maximize this probability, that is the maximum P(Y | X). The algorithm is made possible using Bayes Theorem, a theorem providing a means of calculating an event's probability based on its prior probability. The theorem is as follows:

$$P(Y|X) = \frac{P(X|Y)P(Y)}{P(X)}$$

An important assumption in the Naïve Bayes model is that all features are independent. This allows one to make classifications with much fewer parameters and reduce complexity by a large factor. The calculation of the posterior for each data point allows the algorithm to effectively model the probability distribution of each class.

The Naïve Bayes model performs well in small data sets, often outperforming discriminative due to possible overfitting by classifiers such as SVM. The naïve assumption that features are independent allows Naïve Bayes to make efficient predictions and stay relatively simple compared to other generative models. Unfortunately, the naïve assumption often results in inaccurate probability estimation. This makes Naïve Bayes an unlikely candidate for regression.

Conclusion

Machine learning models can be used in classification tasks to predict a target Y for an input X. Discriminative models do this by modeling the conditional probability distribution P(Y|X). Generative models will take on a more complex task and learn the joint probability distribution P(X,Y). There are different use cases that are optimized by using either a discriminative or generative model. Therefore, it is beneficial to know the advantages and disadvantages of a given model based on its membership to one of the two classes.

References

[1] Applications of Machine Learning - Javatpoint. Retrieved from https://www.javatpoint.com/applications-of-machine-learning

[2] Melvin M. Vopson Senior Lecturer in Physics. (2021, May 04). The world's data explained: How much we're producing and where it's all stored. Retrieved from https://theconversation.com/the-worlds-data-explained-how-much-were-producing-and-where-its-all--stored-159964

[3] What are some benefits and drawbacks of discriminative and generative models? Retrieved from https://www.quora.com/What-are-some-benefits-and-drawbacks-of-discriminative-and-generative-models

[4] Support Vector Machines: A Simple Explanation. (n.d.). Retrieved from https://www.kdnuggets.com/2016/07/support-vector-machines-simple-explanation.html