Naive Bayes Classifier

CS456 - MACHINE LEARNING SPRING 2023

Kaling Vikram Singh & S Sunil Raja

School of Computer Sciences
National Institute of Science Education and Research (NISER)
Bhubaneswar, India

April 21, 2023

Part I: Introduction

1 Introduction 4

2 Bayes' Theorem 5

2.1 Proof 6

2.2 Theory 7

3 Naive Bayes Classifer 9

3.1 Types 9

3.2 Advantages and Disadvantages 10

3.3 Application 11

Part II: Demonstration

1 Code **13**
1.1 Computing prior probabilities 14
1.2 Computing conditional probabilities 15
1.3 Predict class(Applying Bayes Theorem) 16
1.4 Calculating Accuracy/Performance 17

2 Summary **18**

Part I

Naive Bayes Classifier

Introduction

- Naive Bayes is a probabilistic classification algorithm based on Bayes' theorem. It is a simple and effective algorithm.
- It's "naive" as it assumes that the features are independent given the class label. Essentially, the algorithm assumes that the existence or non-existence of a particular feature does not impact the likelihood of another feature's existence or non-existence.*Naive Bayes Classifier in Machine Learning - Javatpoint* 2023

Bayes' Theorem

Bayes' theorem, also known as Bayes' rule or Bayes' law, is a mathematical formula used for calculating conditional probabilities. It describes the probability of an event based on prior knowledge of the conditions that might be related to the event.

Theorem 1 (Bayes' Theorem)

The probability of an event A occurring, given that event B has occurred, is equal to the product of the probability of event B occurring given that event A has occurred, and the prior probability of event A, divided by the prior probability of event B.

$$P(A|B) = \frac{P(B|A) \times P(A)}{P(B)} \tag{1}$$

Let A and B be two events such that P(B) > 0. Then, by the definition of conditional probability:

$$P(A \cap B) = P(A|B) \times P(B)$$

Similarly,

$$P(B \cap A) = P(B|A) \times P(A)$$

Comparing the RHS of both equations;

$$P(A|B) = \frac{P(B|A) \times P(A)}{P(B)}$$

Bayes' Theorem
Theory

Suppose we have a data set D consisting of n instances, where each instance $\vec{x} = (\vec{x_1}, \vec{x_2}, ..., \vec{x_m})$ is a vector of m features and we have a set of n labels $\vec{y}$=(y_1,y_2, ...,y_m). Now,

$$P(\vec{x}|y) = P(\vec{x_1}, \vec{x_2}, ..., \vec{x_m}|y)$$

Assuming that each feature is independent of the other

$$P(\vec{x}|y) = P(\vec{x_1}|y) \times P(\vec{x_2}|y) \times ... \times P(\vec{x_m}|y)$$

This gives the probability of a label y to correspond to a $\vec{x_i}$ is,

$$P(y|\vec{x_i}) = \frac{P(\vec{x_i}|y) \times P(y)}{P(\vec{x_i}}$$

Bayes' Theorem

Theory

To estimate the likelihood probabilities P(x_i | y) for each feature xi and class label y, we can use the maximum likelihood estimator:

$$P(x_i|y) = \frac{\text{number of instances in the training data that have feature } x_i \text{ and class label } y}{\text{number of instances in the training data that have class label y}}$$

Then assuming that

$$P(y|\vec{x}) = \frac{P(y)\prod_{i=1}^{m} P(\vec{x_i}|y)}{P(\vec{x})} 1.9.NaiveBayes2023$$

Naive Bayes Classifer

Types

- Gaussian Naive Bayes: This assumes that the features follow a Gaussian distribution. It is typically used for continuous data (image processing).
- Multinomial Naive Bayes: Used to classify discrete data, such as text classification or spam filtering.
- Bernoulli Naive Bayes: Similar to the Multinomial Naive Bayes classifier, However, data, in this case, is Binary. Used generally for sentiment analysis or fraud detection.

Naive Bayes Classifer

Advantages and Disadvantages

Advantages:

- Computationally efficient and can handle large datasets with high-dimensional features.*What is Naıve Bayes | IBM* 2023
- Relatively easy to implement and can be trained quickly with a small amount of data
- It can handle noisy or incomplete data well. As that feature can be ignored or assigned a lower weight to improve accuracy.

Disadvantages:

- May not work well for datasets with non-linear relationships between features and labels.
- Might require feature scaling or normalization to improve performance.
- Outliers may lead to overfitting or underfitting of the data.

- Spam Filtering: Bayes classifier can be used to classify emails as spam or non-spam based on the text content of the email.*Naive Bayes Classifier in Machine Learning - Javatpoint* 2023
- Medical Diagnosis: Bayes classifier can be used to diagnose medical conditions based on the patient's symptoms and medical history.
- Fraud Detection: Bayes classifier can be used to detect fraudulent transactions by analyzing the transaction data.
- Recommendation Systems: Bayes classifier can be used to recommend products to users based on their past purchases or browsing history.
- Face Recognition: Bayes classifier can be used to classify images of faces based on the features of the face, such as the position of the eyes, nose, and mouth.

Part II

Demonstration

CODE

STEPS INVOLVED

The steps involved in the Naive Bayes classifier are as follows

- Compute the prior probabilities: Calculate the probability of each class occurring in the training data by dividing the number of examples in each class by the total number of examples.
- Compute the conditional probabilities: For each feature in the training data, calculate the probability of that feature occurring in each class separately.
- Apply Bayes' theorem: For each input example, calculate the posterior probability of it belonging to each class by multiplying the prior probability of that class with the conditional probabilities of each feature in the input belonging to that class.
- Evaluate the performance of the model. The class with the highest posterior probability is the predicted class.

Code

Computing prior probabilities

- np.bincounts returns the number of each label in Y train
- priors is a list that contains the probability of finding an instance of a particular label

```
class_counts = np.bincount(Y_train)
priors = class_counts / len(Y_train)
No_of_features=len(features)
```

✓ 0.1s

Figure. Computing Prior Probability

CODE
COMPUTING CONDITIONAL PROBABILITIES

- Initiate a empty array to store the conditional probability.
- For each label and x_i compute the conditional Probability

```python
conditional_probs = {}
for feature in range(0,No_of_features):
    for label in np.unique(Y_train):
        feature_given_label = X_train[Y_train == label, feature]
        conditional_probs[(feature, label)] = len(feature_given_label) / class_counts[label]
```

Figure. Computing Conditional Probability

Code

Predict class(Applying Bayes Theorem)

- Here Multinomial Naive Bayes algorithm is applied, and likelihood is calculated for each unique label.
- The formula discussed above is applied and the probability is found.
- np.argmax returns the index of the label with the highest likelihood or probability.

```
def predict_class(input_data):
    posteriors = []
    for label in np.unique(Y_train):
        likelihood = 1.0
        for feature in range(0,No_of_features):
            likelihood *= conditional_probs[(feature, label)] ** input_data[feature]
        posterior = priors[label] * likelihood
        posteriors.append(posterior)
    return np.unique(Y_train)[np.argmax(posteriors)]
```
✓ 0.0s

Figure. Predicting Class

Code

Calculating Accuracy/Performance

- Returns the accuracy of the classifier.

```python
correct = 0
for i in range(len(X_test)):
    input_data = X_test[i]
    true_label = Y_test[i]
    predicted_label = predict_class(input_data)
    if true_label == predicted_label:
        correct += 1
accuracy = correct / len(X_test)
print(f"Accuracy: {accuracy}")
```

✓ 0.3s

Figure. Computing Accuracy

SUMMARY

- Naives Bayes classifier works on the principle of Bayes theorem and is less computationally intensive.
- It has applications in filtering spam emails and website recommendations.
- One can alternatively use GaussianNb from sklearn.naive_bayes and use the model to fit data.
- A dataset was obtained from Kaggle*Naive bayes classification data* 2023 and the program was run for it (available here).

REFERENCES I

1.9. Naive Bayes (Apr. 2023). [Online; accessed 21. Apr. 2023]. URL: https://scikit-learn.org/stable/modules/naive_bayes.html.

Naive bayes classification data (Apr. 2023). [Online; accessed 21. Apr. 2023]. URL: https://www.kaggle.com/datasets/himanshunakrani/naive-bayes-classification-data?datasetId=2376070.

Naive Bayes Classifier in Machine Learning - Javatpoint (Apr. 2023). [Online; accessed 21. Apr. 2023]. URL: https://www.javatpoint.com/machine-learning-naive-bayes-classifier.

What is Naıve Bayes | IBM (Apr. 2023). [Online; accessed 21. Apr. 2023]. URL: https://www.ibm.com/topics/naive-bayes.