Naive Bayes

School of Data and Computer Science
Sun Yat-sen University

Overview

1. Naive Bayes
2. Datasets
3. Task

1 Naive Bayes

2 Datasets

3 Task

Introduction

- Naive Bayes is a simple technique for constructing classifiers.
- Naive Bayes assign class labels to problem instances, represented as vectors of feature values.
- All naive Bayes classifiers assume that **the value of a particular feature is independent of the value of any other feature**

An Example

Example 1

- For example, a fruit may be considered to be an apple if it is red, round, and about 10 cm in diameter.
- A naive Bayes classifier considers each of these features to contribute independently to the probability that this fruit is an apple, regardless of any possible correlations between the color, roundness, and diameter features.

Naive Bayes Methods

- Naive Bayes methods are a set of supervised learning algorithms based on applying Bayes theorem with the "naive" assumption of conditional independence between every pair of features given the value of the class variable.
- Bayes theorem states the following relationship, given class variable y and dependent feature vector x_1 through x_n:

$$P(y \mid x_1, ..., x_n) = \frac{P(y)P(x_1, ..., x_n \mid y)}{P(x_1, ..., x_n)}$$

Naive Bayes Methods (cont'd)

- Using the naive conditional independence assumption that

$$P(x_i \mid y, x_1, \ldots, x_{i-1}, x_{x+1}, \ldots, x_n) = P(x_i \mid y)$$

- For all i, this relationship is simplified to

$$P(y \mid x_1, \ldots, x_n) = \frac{P(y) \prod_{i=1}^{n} P(x_i \mid y)}{P(x_1, \ldots, x_n)}$$

Naive Bayes Methods (cont'd)

- Since $P(x_1, ..., x_n)$ is constant given the input, we can use the following classification rule:

$$P(y \mid x_1, ..., x_n) \propto P(y) \prod_{i=1}^{n} P(x_i \mid y)$$

$$\hat{y} = \arg\max_y P(y) \prod_{i=1}^{n} P(x_i \mid y),$$

- and we can use Maximum A Posteriori (MAP) estimation to estimate $P(y)$ and $P(x_i \mid y)$, the former is then the relative frequency of class y in the training set.

Naive Bayes Methods (cont'd)

The different naive Bayes classifiers differ mainly by the assumptions they make regarding the distribution of $P(x_i \mid y)$.

- When attribute values are discrete, $P(x_i \mid y)$ can be easily computed according to the training set.
- When attribute values are continuous, an assumption is made that the values associated with each class are distributed according to Gaussian i.e., Normal Distribution.

Naive Bayes Methods (cont'd)

Example 2

For example, suppose the training data contains a continuous attribute x. We first segment the data by the class, and then compute the mean and variance of x in each class. Let μ_k be the mean of the values in x associated with class y_k, and let σ_k^2 be the variance of the values in x associated with class y_k. Suppose we have collected some observation value x_i. Then, the probability distribution of x_i given a class y_k, $P(x_i \mid y_k)$ can be computed by plugging x_i into the equation for a Normal distribution parameterized by μ_k and σ_k^2. That is,

$$P(x = x_i \mid y = y_k) = \frac{1}{\sqrt{2\pi\sigma_k^2}} e^{-\frac{(x_i - \mu_k)^2}{2\sigma_k^2}}$$

Multinomial Naive Bayes

- Multinomial Naive Bayes implements the naive Bayes algorithm for multinomially distributed data, and is one of the two classic naive Bayes variants used in text classification.
- The data are typically represented as word vector counts, although tf-idf vectors are also known to work well in practice.
- The distribution is parametrized by vectors $\theta_y = (\theta_{y1}, ..., \theta_{yn})$ for each class y, where n is the number of features and θ_{yi} is the probability $P(x_i \mid y)$ of feature i appearing in a sample belonging to class y.

Complement Naive Bayes

- Complement Naive Bayes implements the complement naive Bayes (CNB) algorithm.
- CNB is an adaptation of the standard multinomial naive Bayes (MNB) algorithm suited for imbalanced data sets.
- CNB uses statistics from the complement of each class to compute the model's weights.
- The inventors of CNB show empirically that the parameter estimates for CNB are more stable than those for MNB.
- CNB regularly outperforms MNB on text classification tasks.

Bernoulli Naive Bayes

- Bernoulli Naive Bayes implements the naive Bayes training and classification algorithms for data that is distributed according to multivariate Bernoulli distributions.
- There may be multiple features but each one is assumed to be a binary-valued (Bernoulli, boolean) variable.
- This class requires samples to be represented as binary-valued feature vectors
- If handed any other kind of data, a BernoulliNB instance may binarize its input (depending on the binarize parameter).

Categorical Naive Bayes

- Categorical Naive Bayes implements the categorical naive Bayes algorithm for categorically distributed data.
- It assumes that each feature, which is described by the index , has its own categorical distribution.
- For each feature i in the training set X, it estimates a categorical distribution for each feature i of X conditioned on the class y.
- The index set of the samples is defined as $J = \{1, ..., m\}$, with m as the number of samples.

The Naive Bayes Classifier in MATLAB

The naive Bayes classifier is designed for use when predictors are independent of one another within each class, but it appears to work well in practice even when that independence assumption is not valid. It classifies data in two steps:

1. Training step: Using the training data, the method estimates the parameters of a probability distribution, assuming predictors are conditionally independent given the class.
2. Prediction step: For any unseen test data, the method computes the posterior probability of that sample belonging to each class. The method then classifies the test data according the largest posterior probability.

The Naive Bayes Classifier in MATLAB (cont'd)

- Naive Bayes classifiers leverage Bayes theorem and make the assumption that predictors are independent of one another within each class.
- The classifiers appear to work well even when the independence assumption is not valid.
- You can use naive Bayes with two or more classes in Classification Learner.
- The app allows you to train a Gaussian naive Bayes model or a kernel naive Bayes model individually or simultaneously.

Train Classification Models in Classification Learner App

- You can use Classification Learner to train models of these classifiers: decision trees, discriminant analysis, support vector machines, logistic regression, nearest neighbors, naive Bayes, and ensemble classification.
- In addition to training models, you can explore your data, select features, specify validation schemes, and evaluate results.
- You can export a model to the workspace to use the model with new data or generate MATLAB® code to learn about programmatic classification.

Train Classification Models in Classification Learner App (cont'd)

Training a model in Classification Learner consists of two parts:

- Validated Model: Train a model with a validation scheme. The app protects against overfitting by applying cross-validation.
- Full Model: Train a model on full data without validation. The app trains this model simultaneously with the validated model.

1 Naive Bayes

2 Datasets

3 Task

Datasets

- The UCI dataset (`http://archive.ics.uci.edu/ml/index.php`) is the most widely used dataset for machine learning.
- If you are interested in other datasets in other areas, you can refer to `https://www.zhihu.com/question/63383992/answer/222718972`.
- Today's experiment is conducted with the **Adult Data Set** which can be found in `http://archive.ics.uci.edu/ml/datasets/Adult`.
- `https://ww2.mathworks.cn/help/stats/naive-bayes-classification.html`

Adult Data Set

Data Set Characteristics:	Multivariate	**Number of Instances:**	48842	**Area:**	Social
Attribute Characteristics:	Categorical, Integer	**Number of Attributes:**	14	**Date Donated**	1996-05-01
Associated Tasks:	Classification	**Missing Values?**	Yes	**Number of Web Hits:**	1305515

- There are 3 related files: `adult.name` is the description of **Adult Data Set**, `adult.data` is the training set, and `adult.test` is the testing set.
- There are 14 attributes in this dataset.
- Prediction task is to determine whether a person makes over 50K a year.

1 Naive Bayes

2 Datasets

3 Task

Task

- Given the training dataset `adult.data` and the testing dataset `adult.test`, please accomplish the prediction task to determine whether a person makes over 50K a year in `adult.test` by using Naive Bayes algorithm (MATLAB), and compute the accuracy.
- Note: keep an eye on the discrete and continuous attributes.
- Please finish the experimental report.

Reference

- https://scikit-learn.org/stable/modules/naive_bayes.html
- The UCI dataset (http://archive.ics.uci.edu/ml/index.php.
- https://ww2.mathworks.cn/help/stats/Train-Naive-Bayes-Classifiers-Using-Classification-Lea html
- https://ww2.mathworks.cn/help/stats/naive-bayes-classification.html

The End