UECS3213 / UECS3453 Data Mining

SESSION: January, 2019

Lab 8: Naïve Bayes Implementation using scikit-learn

Introduction

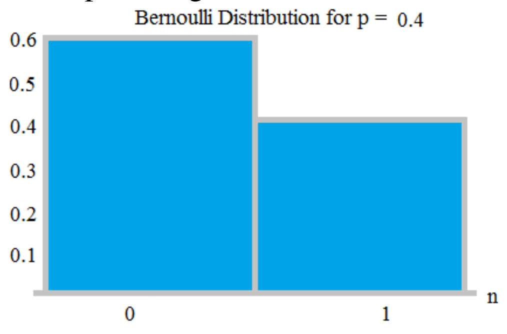
Scikit-learn is a free software machine learning library for the Python programming language.[3] It features various classification, regression and clustering algorithms including support vector machines, random forests, gradient boosting, k-means and DBSCAN, and is designed to interoperate with the Python numerical and scientific libraries NumPy and SciPy.

In this tutorial, you are going to learn how to **build** and **evaluate** a Naive Bayes Classifier using Python's Scikit-learn package. Naive Bayes is the most straightforward and fast classification algorithm, which is suitable for a large chunk of data. Naive Bayes classifier is successfully used in various applications such as spam filtering, text classification, sentiment analysis, and recommender systems. It uses Bayes theorem of probability for prediction of unknown class.

Find the correct distribution function

How to model the probability functions? Scikit-learn provide three naive Bayes implementations: Bernoulli, multinomial and Gaussian. The only difference is about the probability distribution adopted.

- **Bernoulli**: The binomial model is useful if your features are binary.
  - In probability theory and statistics, the Bernoulli distribution, named after Swiss mathematician Jacob Bernoulli, is the discrete probability distribution of a random variable which takes the value 1 with probability p and the value 0 with probability q = 1 − p , that is, the probability distribution of any single experiment that asks a yes–no question; the question results in a boolean-valued outcome, a single bit of information whose value is success/yes/true/one with probability p and failure/no/false/zero with probability q. (https://en.wikipedia.org/wiki/Bernoulli_distribution)

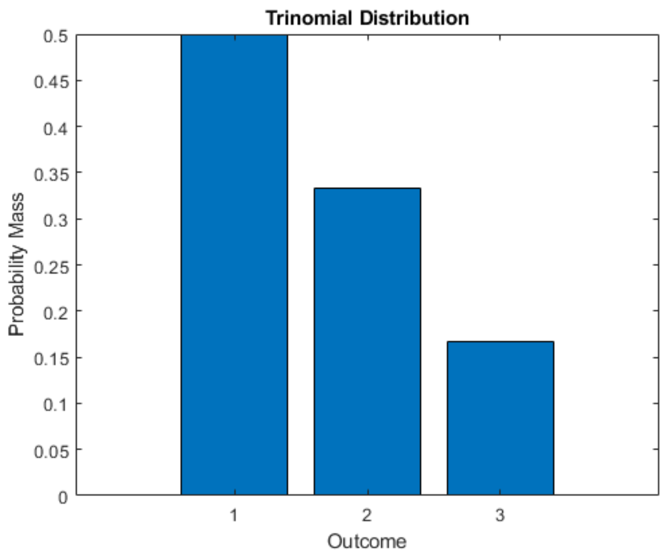


- **BernoulliNB** implements the naive Bayes training and classification algorithms for data that is distributed according to multivariate Bernoulli distributions; i.e., there may be multiple features but each one is assumed to be a binary-valued (Bernoulli, boolean) variable.

$$P(X) = \begin{cases} p & if\ X = 1 \\ q & if\ X = 0 \end{cases}$$

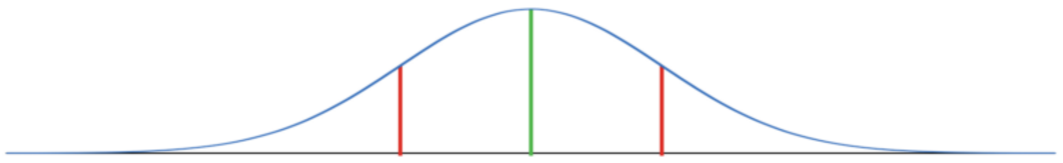
$$where\ q = 1 - p\ and\ 0 < p < 1$$

- **Multinomial**: It is useful if your features are discrete.



  - **MultinomialNB** implements the naive Bayes algorithm for multinomially distributed data, and is one of the two classic naive Bayes variants used in text classification (where the data are typically represented as word vector counts.

- **Gaussian**: It assumes that continuous features follow a normal distribution. It is suitable for more generic classification tasks.



  - **GaussianNB** implements the Gaussian Naive Bayes algorithm for classification. The likelihood of the features is assumed to be Gaussian:

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

For more, read https://scikit-learn.org/stable/modules/naive_bayes.html

Objectives

At the end of this lab, you are expected to acquire the following:

a) Classification Workflow
b) What is Naive Bayes classifier?
c) How Naive Bayes classifier works?
d) Classifier building in Scikit-learn
e) Zero Probability Problem
f) It's advantages and disadvantages

Instruction

1. Visit the "Naive Bayes Classification using Scikit-learn" at the following link: https://www.datacamp.com/community/tutorials/naive-bayes-scikit-learn

2. Follow the step-by-step instructions in the tutorial.

Other Related References

- https://scikit-learn.org/stable/modules/naive_bayes.html
- https://medium.com/@awantikdas/a-comprehensive-naive-bayes-tutorial-using-scikit-learn-f6b71ae84431
- https://www.datacamp.com/courses/supervised-learning-with-scikit-learn
- https://hub.packtpub.com/implementing-3-naive-bayes-classifiers-in-scikit-learn/
- https://stackabuse.com/the-naive-bayes-algorithm-in-python-with-scikit-learn/
- https://blog.sicara.com/naive-bayes-classifier-sklearn-python-example-tips-42d100429e44

The End