Journal of Machine Learning Research 1 (2019) INSERT PAGE STUFF Submitted 9/19; Published DATE PUBLISHED

Learning with Naive Bayes

Andrew Kirby ANDY-KIRBY@LIVE.COM
Kevin Browder BROWDERKEVIN54@GMAIL.COM
Nathan Stouffer NATHANSTOUFFER1999@GMAIL.COM
Eric Kempf ERICKEMPF123@GMAIL.COM

Editor: Kevin Browder

Abstract

Keywords:

1. Introduction

2. Problem Statement

The problem that we must solve is a classification problem. Given an input file that contains examples (each example consists of a list of attributes and an associated classification), our task is to implement a learning algorithm that is trained to classify examples. The algorithm we will implement is called Naive Bayes. The performance of our learning algorithm will be evaluated by two metrics of our choosing and 10-fold cross validation. When we have implemented the algorithm, we then perform our experiment. We are tasked with testing whether scrambling values in 10% of the features will affect the performance of Naive Bayes. This effectively eliminates the usefulness of 10% of features in a given data set.

2.1 Hypothesis

We predict that scrambling 10% of features will marginally affect performance.

3. Algorithm

-It be what it do

4. Experimental Design

4.1 Pre-processing

The pre-processing is done using the pandas library in Python. The class column is moved before all of the attributes for a consistent output between datasets. Next the examples are randomly shuffled. A new column is added after the class column and each example is assigned to one of 10 sets that are used in the 10 fold cross validation. If the data is continuous, it is discretized into 5 bins with each bin containing an equal number of

examples. Non numerical data is changed to integers because the algorithm can only read integers, this includes the class names. Before output three lines of data are generated, the first line includes the number of classes, number of attributes and the number of examples. The second line includes the number of bins for each attribute. The third line include the class names so the algorithm can convert back from numerical class names to the original string names. After these lines are generated they are outputted to the first three lines of the .csv file and the pre-processed data is outputted starting on the fourth line. After this data is outputted ten percent of the attributes are randomly chosen and the data in each of these attributes is randomly shuffled. This shuffled data is then outputted to a new .csv.

4.2 Tuning

Add stuff about bin size, what else?
Experimenting w/ using different attributes

4.3 Final Parameters

-Bin size
-Number of attributes used
-Which attributes used?

5. Results

6. Summary

References

Dheeru Dua and Casey Graff. UCI machine learning repository, 2017. URL `http://archive.ics.uci.edu/ml`.