Data Mining Coursework

Outline Solution

https://machinelearningmastery.com/use-ensemble-machine-learning-algorithms-weka/ (voting commitee)

**Abstract**

In this report we provide a solution to a….. Using data from a marketing campaigns of a banking institution…

**1. Introduction**

This document presents an outline solution to the coursework. It is not meant to be a complete report, but highlights some of the issues and results that arise when following a systematic data mining process.

The goal is to develop models to classify…. under both equal and unequal costs. In the following sections, we report on data exploration, data pre-processing, model building, and results.

…

**2. Data exploration**

The dataset was supplied as an ARFF file ready for use with the Weka data mining toolkit. Initial exploration of the training dataset showed the following features.

* The 'no' class made up roughly 88% of the data. This is the accuracy of the default classifier
* The data set has no missing values.
* The data set has 17 attributes and 36,000 instances. Since we have a large amount of data, reducing the number of attributes might not be necessary. We would try to reduce the data set anyway…
* Some attributes have imbalanced distributions of values:
  + Default: More than 95% of the data have value ‘yes’
  + Loan: Most of the data have value ‘no’
* Looking at the histograms of each attribute, there were no attribute that were strongly predictive of the class. *Viewing the histograms for each variable showed that there were no variables that were strongly predictive of the class*. (In the histogram of the 'default' attribute, it might look like that a 'yes' label of this attribute would belong to class 'no'. But inspecting the scatter plot showing that there are small number of instances with label 'yes' belong to class 'yes'

**3. Data pre-processing**

There are 7 numeric attributes: age (a1), balance (a6), day (a10), duration (a12), campaign (a13), pdays (a14) and previous (a15).

Principal component for in attribute selection to plot scatter plot.  
add expression

**4. Classification Models**

**4.1 Benchmark models**

As basic benchmarks on the dataset, naive Bayes, k-nearest neighbour, logistic regression and J4.8 were applied without any pre-processing and with their default parameters. 10-fold cross-validation was used to increase the reliability of the error estimate. The value of k was chosen by cross-validation and the value selected was 9.

|  |  |
| --- | --- |
| Model | Accuracy |
| naive Bayes | 87.8 % |
| k-nearest neighbour (k = 9) | 88.9 % |
| logistic regression | 90.0 % |
| J4.8 | 90.3 % |

**4.2 Attribute selection**

Performing attribute selection algorithms on the data set give us these results.

**4.3 Model development**

**4.3.1 Naïve Bayes**

This model performed consistently well, but was at its best with the full dataset. The main issue to explore is the representation of numeric variables: some have few values (so don’t look Gaussian), for example installment commitment, while others are skew, such as credit amount. There are two ways to address this: either to use a kernel density estimator to approximate the non-Gaussian distributions, or to discretise the attributes. We chose the second alternative for simplicity. Attributes 18–20 were removed on the basis that they had very little predictive power. Those attributes with a small number of values were discretised into the same number of bins; those with a large number of values were discretised into 10 bins of equal frequency. The naive Bayes model trained on this dataset had an accuracy of 74.2%, which is little different from the benchmark.

After discretising 88.4981 % 10 bins of equal frequency: false

88.1553 % 10 bins of equal frequency: true

**4.3.2 k-nearest Neighbour**

Normalise data

88.8741 % standard

88.7304 % 1/distance

88.8741 % 1 - distance

**4.3.3 Logistic regression**

90.0354 % 1 × 10−8

90.0354 % 1 × 10−4

90.0354 % 1.0

90.0464 % 10

**4.3.4 Decision Trees**

Post-pruning 0.35 89.7478 %

Post-pruning 0.30 89.9469 %

Post-pruning 0.25 74.8%

Post-pruning 0.2 90.4557 %

Post-pruning 0.15 90.4778 %

Post-pruning 0.10 90.2013 %

Reduced error pruning — 90.1902%

**4.4 Combining Models**

**5. Evaluation and Conclusions**

The maximum achieved by anyone was 82%. The test dataset contains 34 good examples and 16 bad examples, so the default rule (classifying every example as good) has an accuracy of 68%

The performance of….

% Rows Columns 2 2

% Matrix elements

0.0 1.0

10.0 0.0