Machine Learning Project

Research · May 2016

DOI: 10.13140/RG.2.1.3176.2800

Manish Bhatt
University of New Orleans

CSCI 4525 Project IV: Machine Learning Project

Manish Bhatt

Student ID: 2451137

Department of Computer Science

Submitted to: Dr. Stephen Ware, PhD

Narrative Intelligence Lab

Department of Computer Science

University of New Orleans, New Orleans, LA, 70148.

Abstract

In this project, we were asked to experiment with a real world dataset, and to explore how machine learning algorithms can be used to find the patterns in data. We were expected to gain experience using a common data-mining and machine learning library, Weka, and were expected to submit a report about the dataset and the algorithms used. After performing the required tasks on a dataset of my choice, herein lies my final report.

Keywords: Machine Learning, Pattern Recognition, Classification, Supervised learning, Artificial Intelligence.

CSCI 4525 Project IV: Machine Learning Project

Introduction:

Machine learning is a sub-domain of computer science which evolved from the study of pattern recognition in data, and also from the computational learning theory in artificial intelligence. It is the first-class ticket to most interesting careers in data analytics today[1]. As data sources proliferate along with the computing power to process them, going straight to the data is one of the most straightforward ways to quickly gain insights and make predictions.

Machine Learning can be thought of as the study of a list of sub-problems, viz: decision making, clustering, classification, forecasting, deep-learning, inductive logic programming, support vector machines, reinforcement learning, similarity and metric learning, genetic algorithms, sparse dictionary learning, etc. Supervised learning, or classification is the machine learning task of inferring a function from a labeled data [2]. In Supervised learning, we have a training set, and a test set. The training and test set consists of a set of examples consisting of input and output vectors, and the goal of the supervised learning algorithm is to infer a function that maps the input vector to the output vector with minimal error. In an optimal scenario, a model trained on a set of examples will classify an unseen example in a correct fashion, which requires the model to generalize from the training set in a reasonable way. In layman’s terms, supervised learning can be termed as the process of concept learning, where a brain is exposed to a set of inputs and result vectors and the brain learns the concept that relates said inputs to outputs. A wide array of supervised machine learning algorithms are available to the machine learning enthusiast, for example Neural Networks, Decision Trees, Support Vector Machines, Random Forest, Naïve Bayes Classifier, Bayes Net, Majority Classifier[4,7,8,9] etc., and they each have their own merits and demerits. There is no single algorithm that works for all cases, as merited by the *No free lunch theorem* [3]. In this project, we try and find patterns in a dataset [2], which is a sample of males in a heart-disease high risk region of South Africa, and attempt to throw various intelligently-picked algorithms at the data, and see what sticks.

Problems and Issues in Supervised learning:

Before we get started, we must know about how to pick a good machine learning algorithm for the given dataset. To intelligently pick an algorithm to use for a supervised learning task, we must consider the following factors [4]:

1. Heterogeneity of Data:

   Many algorithms like neural networks and support vector machines like their feature vectors to be homogeneous numeric and normalized. The algorithms that employ distance metrics are very sensitive to this, and hence if the data is heterogeneous, these methods should be the afterthought. Decision Trees can handle heterogeneous data very easily.

2. Redundancy of Data:

   If the data contains redundant information, i.e. contain highly correlated values, then it’s useless to use distance based methods because of numerical instability. In this case, some sort of Regularization can be employed to the data to prevent this situation.

3. Dependent Features:

   If there is some dependence between the feature vectors, then algorithms that monitor complex interactions like Neural Networks and Decision Trees fare better than other algorithms.

4. Bias-Variance Tradeoff:

   A learning algorithm is biased for a particular input x if, when trained on each of these data sets, it is systematically incorrect when predicting the correct output for x, whereas a learning algorithm has high variance for a particular input x if it predicts different output values when trained on different training sets. The prediction error of a learned classifier can be related to the sum of bias and variance of the learning algorithm, and neither can be high as they will make the prediction error to be high. A key feature of machine learning algorithms is that they are able to tune the balance between bias and variance automatically, or by manual tuning using bias parameters, and using such algorithms will resolve this situation.

5. Curse of Dimensionality:

   If the problem has an input space that has a large number of dimensions, and the problem only depends on a subspace of the input space with small dimensions, the machine learning algorithm can be confused by the huge number of dimensions and hence the variance of the algorithm can be high. In practice, if the data scientist can manually remove irrelevant features from the input data, this is likely to improve the accuracy of the learned function. In addition, there are many algorithms for feature selection that seek to identify the relevant features and discard the irrelevant ones, for instance **Principle Component Analysis** for unsupervised learning. This reduces the dimensionality.

6. Overfitting:

   The programmer should know that there is a possibility that the output values may constitute of an inherent noise which is the result of human or sensor errors. In this case, the algorithm must not attempt to infer the function that exactly matches all the data. Being too careful in fitting the data can cause overfitting, after which the model will answer perfectly for all training examples but will have a very high error for unseen samples. A practical way of preventing this is stopping the learning process prematurely, as well as applying filters to the data in the pre-learning phase to remove noises.

Only after considering all these factors can we pick a supervised learning algorithm that works for the dataset we are working on. For example, if we were working with a dataset consisting of heterogeneous data, then decision trees would fare better than other algorithms. If the input space of the dataset we were working on had 1000 dimensions, then it's better to first perform PCA on the data before using a supervised learning algorithm on it.

Dataset:

The dataset used is a sample of males in a heart-disease high-risk region of the Western Cape, South Africa. The dataset that was used for this project is a subset of a much larger dataset, as described in Rousseauw et al, 1983, South African Medical Journal, and has the following feature vectors:

1.	sbp	systolic blood pressure
2.	tobacco	cumulative tobacco (kg)
3.	ldl	low density lipoprotein cholesterol
4.	adiposity	this is the amount of fat tissue in the body
5.	famhist	family history of heart disease (Present, Absent) (String)
6.	typea	type-A behavior
7.	obesity	State of being overweight

8. alcohol	current alcohol consumption
9. age	Age at onset
10. chd	coronary heart disease (Class Label of the Dataset)

In the dataset, there are 462 example vectors. Expert Systems have been used in the field of medical science to assist the doctors in making certain diagnoses, and this can help save lives. Coronary Heart Disease is a disease where a waxy substance builds up inside the coronary arteries, and hence this may lead to heart attack, and even death. When diagnosed and treated, the treatment can go a long way in helping the patient. This classification task is important because the expert system, when correctly generalized, can tell the doctor which patient may have the disease, and the doctor can take a look at that case in more detail. Moreover, if the doctor makes a slip, i.e. misdiagnoses someone, the expert system can help rectify his mistake. It results in two doctors, one of them virtual, instead of one doctor diagnosing every case which has a greater chance of accuracy and precision.

First we perform the significance analysis of the 9 feature vectors, to see which vectors have more significance in representing the classes. We used **Principal Component Analysis**[4,7,9] for this purpose and came up with the following results.

Attribute Evaluator (supervised, Class (nominal): 10 chd):

Correlation matrix

1	0.21	0.16	0.36	-0.09	-0.06	0.24	0.14	0.39
0.21	1	0.16	0.29	-0.09	-0.01	0.12	0.2	0.45
0.16	0.16	1	0.44	-0.16	0.04	0.33	-0.03	0.31
0.36	0.29	0.44	1	-0.18	-0.04	0.72	0.1	0.63
-0.09	-0.09	-0.16	-0.18	1	-0.04	-0.12	-0.08	-0.24

```
-0.06  -0.01   0.04  -0.04  -0.04   1      0.07   0.04  -0.1
 0.24   0.12   0.33   0.72  -0.12   0.07   1      0.05   0.29
 0.14   0.2   -0.03   0.1   -0.08   0.04   0.05   1      0.1
 0.39   0.45   0.31   0.63  -0.24  -0.1    0.29   0.1    1
```

Ranked attributes:

0.6795   1 0.516adiposity+0.46 age+0.401obesity+0.334ldl+0.324sbp...

0.5465   2 0.543alcohol+0.459tobacco-0.392obesity-0.364ldl-0.282typea...

0.4269   3 -0.792typea-0.459alcohol+0.338famhist+0.135age+0.125sbp...

0.322   4 -0.833famhist-0.305obesity-0.258alcohol-0.21typea-0.196sbp...

0.2291   5 0.624tobacco-0.419alcohol+0.321typea+0.305famhist-0.283obesity...

0.1446   6 0.781sbp-0.379alcohol+0.332typea-0.215ldl-0.174obesity...

0.0706   7 0.788ldl-0.333obesity+0.277alcohol+0.268sbp-0.196adiposity...

0.0194   8 0.691age-0.489tobacco-0.339obesity-0.235sbp+0.187famhist...

Selected attributes: 1,2,3,4,5,6,7,8 : 8

Here we can see that all factors are important after we do the PCA. The last feature has been deemed unworthy by the PCA implementation in WEKA, which made little sense to us as age is highly correlated to most diseases. We further our investigation by using another attribute selector, the **Significance Attribute Evaluator**[5,9], on the data to yield.

Significance feature evaluator

Ranked attributes:

0.301   9 age

0.299   2 tobacco

0.293   6 typea

```
0.269   5 famhist
0.242   3 ldl
0.235   4 adiposity
0.205   1 sbp
0       7 obesity
0       8 alcohol
```

Selected attributes: 9,2,6,5,3,4,1,7,8 : 9

Here, we see that feature 9, i.e. the age of the patient was the most significant factor for classification purposes, and factors 7 and 8, obesity and alcohol consumption were the least significant factors. Through combined results of PCA and SAE, we conclude that all the features are relevant for our purposes. The name of the sample was removed as well. Except for the use of PCA and SAE, no other pre-processing was done on the data.

Baseline Classifier:

As the baseline classifier, we chose a Naïve Bayesian Network because it is easy to compute, and because the features in the given dataset are all aspects of a person's physical habits or medical history, and hence can be assumed to be independent of each other, which is the primary assumption in Naïve Bayes Classifier[6,8,9]. It is a conditional probability model: given a problem instance to be classified, represented by a vector $\mathbf{x} = (x_1, \dots, x_n)$ representing some n features (independent variables), it assigns to this instance probabilities $p(C_k | x_1, \dots, x_n)$ for each of K possible outcomes or classes. The problem with the above formulation is that if the number of features n is large or if a feature can take on a large number of values, then basing such a model on probability tables is infeasible. We therefore reformulate the model to make it more tractable. Using Bayes' theorem, the conditional probability can be decomposed as:

$$p(C_k|\mathbf{x}) = \frac{p(C_k)\ p(\mathbf{x}|C_k)}{p(\mathbf{x})}$$

So, assuming that $p(x_i|x_{i+1},\dots,x_n,C_k) = p(x_i|C_k)$ for i = 1, ….. n-1. So, under the independence assumptions, we can say that

$$p(C_k|x_1,\dots,x_n) = \frac{1}{Z}p(C_k)\prod_{i=1}^{n} p(x_i|C_k)$$

Where, the evidence $Z = p(\mathbf{x})$ is a scaling factor dependent only on $x_1,\dots,x_n$, that is, a constant if the values of the feature variables are known.

Until now, we have derived an independent feature model. In Naïve Bayes classifier, we combine this model with a decision rule, and one of the common rules is to pick which hypothesis is the most probable. The corresponding classifier, a Bayes classifier, is the function that assigns a class label $\hat{y} = C_k$ for some *k* as follows:

$$\hat{y} = \underset{k\in\{1,\dots,K\}}{\operatorname{argmax}}\ p(C_k)\prod_{i=1}^{n} p(x_i|C_k).$$

The Naïve Bayes network is already implemented in the Machine Learning library WEKA[5,9]. This was used on the aforementioned dataset, which led to the following output:

=== Classifier model (full training set) ===

Naive Bayes Classifier

```
                 Class
Attribute            0        1
                (0.65)   (0.35)
===================================
sbp
  mean          135.6278 143.8165
  std. dev.      17.8582  23.5657
  weight sum         302      160
  precision        1.918    1.918

tobacco
  mean            2.6347   5.5268
  std. dev.       3.6078    5.551
  weight sum         302      160
  precision       0.1465   0.1465

ldl
  mean            4.3436    5.489
  std. dev.        1.867   2.2194
  weight sum         302      160
  precision       0.0438   0.0438

adiposity
```

```
  mean          23.9695  28.1196
  std. dev.      7.7602   7.0393
  weight sum        302      160
  precision      0.0878   0.0878

famhist
  Present          97.0     97.0
  Absent          207.0     65.0
  [total]         304.0    162.0

typea
  mean          52.3541  54.4835
  std. dev.      9.5073  10.2082
  weight sum        302      160
  precision      1.2264   1.2264

obesity
  mean          25.7357  26.6251
  std. dev.      4.0833   4.3755
  weight sum        302      160
  precision      0.0799   0.0799

alcohol
```

```
  mean          15.9324  19.1369
  std. dev.     23.4718  26.0986
  weight sum        302      160
  precision      0.5935   0.5935

age
  mean          38.8627  50.3335
  std. dev.     14.8623  10.7993
  weight sum        302      160
  precision      1.0208   1.0208

Time taken to build model: 0 seconds

=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances         331               71.645  %
Incorrectly Classified Instances       131               28.355  %
Kappa statistic                          0.3855
Mean absolute error                      0.3238
Root mean squared error                  0.4725
Relative absolute error                 71.4816 %
Root relative squared error             99.3063 %
```

```
Coverage of cases (0.95 level)         92.4242 %

Mean rel. region size (0.95 level)      79.6537 %

Total Number of Instances              462


=== Detailed Accuracy By Class ===


                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area

Class

                 0.762    0.369    0.796      0.762    0.778      0.386    0.749     0.843     0

                 0.631    0.238    0.584      0.631    0.607      0.386    0.749     0.580     1

Weighted Avg.    0.716    0.324    0.722      0.716    0.719      0.386    0.749     0.752


=== Confusion Matrix ===


   a   b   <-- classified as

 230  72 |   a = 0

  59 101 |   b = 1
```

As we can see that the Naïve Bayes classifier works really well with the given dataset, with the True Positive classification rate being 71.6 percent on an average, i.e. this classifier can correctly classify 71.6 percent of all the examples it sees. However, there is still a vast majority of the dataset, i.e. 28.4% which can’t be correctly classified. This means that our expert medical diagnosis system still misdiagnoses one third of its cases, and one third of the patients’ symptoms who may have the disease are not being scrutinized by the doctor. We will now attempt to improve result by using other more sophisticated classifiers.

Classifier:

Since it's a binary dataset with the class label being either the person has CHD or s/he doesn't have CHD, and the number of samples is less than 100 times the number of features, the correlation matrix shows us that the correlation between various features is under .5, we believe that support vector machines would be a viable classifier in this case. We use the SMO (Sequential Minimal Optimization) algorithm to train support vector machines[7,8,9]. We get the following output:

```
=== Classifier model (full training set) ===
SMO
Kernel used:
  Linear Kernel: K(x,y) = <x,y>
Classifier for classes: 0, 1
BinarySMO
Machine linear: showing attribute weights, not support vectors.
         0.6777 * (normalized) sbp
 +       1.9084 * (normalized) tobacco
 +       1.9409 * (normalized) ldl
 +       0.4788 * (normalized) adiposity
 +      -0.9024 * (normalized) famhist
 +       1.8101 * (normalized) typea
 +      -1.3165 * (normalized) obesity

 +      -0.1267 * (normalized) alcohol

 +       1.3176 * (normalized) age

 -      2.7267


Number of kernel evaluations: 15736 (68.637% cached)


Time taken to build model: 0.02 seconds


=== Stratified cross-validation ===

=== Summary ===


Correctly Classified Instances          328                70.9957 %

Incorrectly Classified Instances        134                29.0043 %

Kappa statistic                         0.3319

Mean absolute error                       0.29

Root mean squared                       0.5386

Relative absolute error                  64.028  %

Root relative squared error             113.1898 %

Coverage of cases (0.95 level)           70.9957 %

Mean rel. region size (0.95 level)       50      %

Total Number of Instances                462
```

```
=== Detailed Accuracy By Class ===

              TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area
Class
              0.825    0.506    0.755      0.825    0.788      0.335    0.659     0.737     0
              0.494    0.175    0.598      0.494    0.541      0.335    0.659     0.471     1
Weighted Avg.    0.710    0.392    0.700      0.710    0.702      0.335    0.659     0.645

=== Confusion Matrix ===

   a   b   <-- classified as
 249  53 |   a = 0
  81  79 |   b = 1
```

Here, we can see that the said SVM performs better than the Naïve Bayes classifier for class 0, predicting 82.5% of the classes correctly, whereas it performs slightly worse than Naïve Bayes for class 1 with 49.4%. On an average, the true positive rate was achieved to be 71% as compared to 71.6% in case of Naïve Bayes. This result is surprising, as we expected SVM to perform better than the Naïve Bayes Classifier for independent non-redundant feature vectors as SVM projects low-dimensional sub-space to a higher dimensional subspace where the features are linearly separable. The RMS error for SVM was comparatively higher compared to Naïve Bayes by .10 and the kappa statistic of Naïve Bayes was lower than SVM by .05, which shows that Naïve Bayes is the better classifier.

Curious about why the data was behaving the way it was, we did use other classifiers on the said dataset. We used Multilayer Perceptron, Decision Tree (J48) [8,9], Random Forest[8,9] with 100 trees, and the only classifier that got close was the J48 with true positive rate of 70.7%. Single Multilayered Perceptron [7,8,9] performed poorly with only 63% TPR, and a deep-learning neural net performed with 65.38% correct classifications. Curious if Lazy learning [8,9] could do any better, we tried it and found that it correctly classified 61.25% of the cases. The only thing we could now think of is that the input space was incomplete, and needed more dimensions for better predictions, and with the given feature vectors,

Conclusion:

We conclude that the dataset is not a complete space, and there are still other feature vectors missing from it. What we were attempting to generalize is a subspace of the actual input space, where the other dimensions are not known, and hence none of the classifiers were able to do better than 71.6% (Naïve Bayes). In the future, if similar studies are conducted to generate the dataset used in this report, more feature vectors need to be calculated so that the classifiers can form a better idea of the problem at hand.

References

[1] "Intro to Machine Learning | Udacity." Intro to Machine Learning | Udacity. Accessed April 27, 2016. https://www.udacity.com/course/intro-to-machine-learning--ud120.

[2] "Elements of Statistical Learning: Data Mining, Inference, and Prediction. 2nd Edition. Datasets:Coronary Heart Disease Dataset." Elements of Statistical Learning: Data Mining, Inference, and Prediction. 2nd Edition. Accessed April 27, 2016. http://statweb.stanford.edu/~tibs/ElemStatLearn/.

[3] "No Free Lunch Theorems." No Free Lunch Theorems. Accessed April 27, 2016. http://www.no-free-lunch.org/.

[4] Hastie, Trevor, Robert Tibshirani, and J. H. Friedman. The Elements of Statistical Learning: Data Mining, Inference, and Prediction: With 200 Full-color Illustrations. New York: Springer, 2001.

[5] "Weka 3: Data Mining Software in Java." Weka 3. Accessed April 27, 2016. http://www.cs.waikato.ac.nz/ml/weka/.

[6] Bozhinova, Monika, Nikola Guid, and Damjan Strnad. Naivni Bayesov Klasifikator: Diplomsko Delo. Maribor: M. Bozhinova, 2015.

[7] Schölkopf, Bernhard, Christopher J. C. Burges, and Alexander J. Smola. Advances in Kernel Methods: Support Vector Learning. Cambridge, MA: MIT Press, 1999.

[8] Norving, Peter, and Stuart Russel. Artificial Intelligence: A Modern Approach. S.l.: Pearson Education Limited, 2013.

[9] Witten, I. H., and Eibe Frank. Data Mining: Practical Machine Learning Tools and Techniques. Amsterdam: Morgan Kaufman, 2005.