**A comparative analysis on classification algorithms in R programming**

**Dataset Details**

|  |  |
| --- | --- |
| Name | Wisconsin Diagnostic Breast Cancer |
| Attributes | 10 |
| Task | Classification |
| Instances | 699 |
| Link | <http://archive.ics.uci.edu/ml/machine-learning-databases/breast-cancer-wisconsin/breast-cancer-wisconsin.data> |
| Type | Multivariate |

**Preprocessing**

* null values are removed
* question marks present in the data are replaced
* Column names are added
* The data is scaled

Pseudocode for preprocessing

*data <- data[-1]*

*data[ is.na(data) ] <- 0*

*data[data == "?"] <- 0*

*apply(data,2,function(x) sum(is.na(x)))*

**Evaluation Metric**

**Precison** evaluation metric is used

**Results Table**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Classifier** | **Nfold Cross Validation** | **Parametres used** | **Accuracy** | **Accuracy**  **Precision** |
| **Decision Tree** | 10 | **Cost factor=0.1**  **depth=10** | **95.184** | **0.949** |
| **Perceptron** | 10 | **Threshold=5**  **activation function=tanh** | **83.23** | **0.635** |
| **NeuralNetworks** | 10 | **Hiddenlayers=3**  **Threshold=0.1** | **95** | **0.502** |
| **Deep Learning** | 10 | **Hiddenlayers=20**  **Threshold=0.2** | **85.9294** | **0.321** |
| **SVM** | 10 | **Cost=10**  **Gamma=0.1**  **Threshold=Linear** | **97.224** | **0.987** |
| **Naivebayes** | 10 | **Na.action=omit/pass** | **97.50** | **0.993** |
| **K-NN** | 10 | **K=5**  **L=2**  **Linear** | **99.56** | **0.997** |
| **AdaBoosting** | 10 | **Iteration=20**  **Delta=4**  **Bagiteration=20**  **Type=gentle** | **96.48** | **0.98** |
| **Bagging** | 10 | **Mfinal=10**  **Length=4**  **Iterations=1000**  **Cp=0** | **96.64** | **0.96** |
| **Logistic Regression** | 10 | **Family=Bipnomial/Quassy** | **44.62** | **0.059** |
| **Gradient Boosting** | 10 | **Ntree=2000**  **Shrinkage=0.02**  **Depth=10**  **Bagfraction=0.1** | **99.56** | **0.98** |
| **RandomForests** | 10 | **Ntree=1000**  **Mty=2**  **Proximity=true**  **Maxfeatures=2** | **96.75** | **0.998** |

**Algorithms**

* Decision Trees
  + Perceptron
  + Neural Net
  + Deep Learning
  + SVM
  + naïve Bayes
  + Logistic Regression
  + k-Nearest Neighbors
  + Bagging
  + Random Forests
  + AdaBoost
  + Gradient Boosting

**Analysis**

Experiments were carried out by providing the same dataset as input to 12 different classifiers in order to assess and compare the performances of algorithms and the effectiveness of the hypothesis created from them. Experiment included studying each of the classifying algorithm, selecting different input parameters for each of them, verifying how each of them influence the output and logging each instance of them. After few comparative dry runs, the input combination that resulted in maximum accuracy was then set as best set of parameters for each of the classifier.

In the goal of highlighting the strength and weaknesses of algorithms being compared, apart from just accuracy or misclassification error as a performance metric, we also included in the comparative study, an additional evaluation parameter “precision” which is usually referred as positive predictive value and gives the preciseness of the trained model in terms of probability of relevancy in the predictions.

Both of these metrics were then collected from repeated runs of cross validation where overall accuracy and precision were averaged across each fold of validation. Such a comparison on the taken breast cancer dataset resulted in the below observations:

**10 fold cross validation on each classifier**

**1)Decision Trees**

|  |  |  |
| --- | --- | --- |
| **Parametre-1**  **Complexity Parametre(cp)** | **Parametre-2**  **Maximum Depth** | **AverageAccuracy** |
| 0.1 | 10 | 91.108 |
| 0.2 | 10 | 91.1058 |
| 0.2 | 30 | 91.10 |
| 0.02 | 30 | 93.287 |
| 0.01 | 20 | 93.865 |
| 0.05 | 20 | 93.135 |
| 0.06 | 20 | 92 |
| 0.3 | 20 | 91.1086 |
| 0.03 | 10 | 93.438 |
| **0..01** | **10** | **95.1843** |
| 0.2 | 5 | 91.1058 |
| 0.2 | 2 | 91.108 |
| 0.1 | 20 | 91.1086 |
| 0.1 | 25 | 91.1086 |
| 0.05 | 20 | 93.1357 |
| 0.5 | 10 | 91.1086 |
| 0.25 | 10 | 91.1086 |

**2)Perceptron**

|  |  |  |
| --- | --- | --- |
| **Parametre1**  **Threshold** | **Parametre-2**  **Learningrate** | **Average Accuracy** |
| 0.01 | Logistic | 82.85 |
| 0.02 | Logistic | 82.85 |
| **5** | **Tanh** | **83.23** |
| 10 | Tanh | 82.54 |
| 20 | Logistic | 82.10 |
| 25 | Tanh | 81.96 |
| 30 | Logistic | 81.667 |
| 25 | Logistic | 81.96 |
| 50 | Tanh | 80.220 |
| 0.01 | Tanh | 82.85 |
| 40 | Logistic | 81.23 |
| 10 | Logistic | 82.54 |
| 20 | Tanh | 82.10 |
| 30 | Tanh | 81.667 |
| 15 | Tanh | 79.75 |

**3)Neural Networks**

|  |  |  |
| --- | --- | --- |
| **Parametre-1**  **hidden layers** | **Parametre2**  **Threshold** | **Accuracy** |
| 3 | 0.01 | 94.89 |
| **3** | **0.1** | **95** |
| 3 | 0.2 | 94.47 |
| 1 | 0.2 | 85.93 |
| 4 | 0.3 | 90.78 |
| 4 | 0.2 | 91.085 |
| 4 | 0.1 | 91.087 |
| 4 | 0.01 | 91.222 |
| 5 | 0.2 | 93.8474 |
| 5 | 0.1 | 94.1674 |
| 5 | 0.3 | 93.8944 |
| 5 | 0.4 | 93.5842 |
| 7 | 0.1 | 91.197 |
| 10 | 0.1 | 93.4199 |

**4)Deep Learning**

|  |  |  |
| --- | --- | --- |
| **Parametre-1**  **hidden layers** | **Parametre-2**  **threshold** | **Accuracy** |
| 50 | 0.1 | 77.1592 |
| 40 | 0.1 | 75.3439 |
| **20** | **0.2** | **85.9294** |
| 40 | 0.2 | 77.3797 |
| 30 | 0.1 | 81.661 |
| 35 | 0.1 | 80.71 |
| 50 | 0.2 | 77.1592 |
| 40 | 0.2 | 75.3439 |
| 35 | 0.2 | 80.43 |
| 45 | 0.2 | 75.75 |
| 45 | 0.1 | 77.269 |
| 25 | 0.1 | 80.9667 |
| 25 | 0.2 | 81.6448 |
| 20 | 0.1 | 85.9294 |

**5)SVM**

|  |  |  |  |
| --- | --- | --- | --- |
| **Parametre-1**  **Cost** | **Parametre-2**  **Gamma** | **Parametre-3**  **Kernel type** | **Average Accuracy** |
| **10** | **0.1** | **Linear** | **97.224875** |
| 30 | 0.3 | Linear | 90.52 |
| 50 | 0.4 | Linear | 93.25 |
| 20 | 0.1 | Polynomial | 96.32 |
| 40 | 0.2 | Polynomial | 87.325 |
| 50 | 0.4 | Polynomial | 91.021 |
| 25 | 0.1 | Radial | 95.32 |
| 35 | 0.3 | Radial | 92.156 |

**6)Naïve Bayes Classifier**

|  |  |
| --- | --- |
| **Parametre-1**  **(na)** | **Accuracy** |
| **Pass** | **97.40** |
| **Omit** | **97.40** |

**7)K-NN**

|  |  |  |
| --- | --- | --- |
| **Parametre-1**  **K** | **Parametre-2**  **L** | **AverageAccuracy** |
| **5** | **2** | **99.85507246** |
| 3 | 2 | 96.223 |
| 10 | 3 | 97.258 |
| 20 | 3 | 95.236 |
| 30 | 2 | 92.564 |
| 5 | 4 | 90.231 |
| 6 | 5 | 87.235 |

**8)RandomForests**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Parametre-1**  **Ntree** | **Parametre-2**  **ntry** | **Parametre-3**  **Proximity** | **Parametre-4**  **Maxfeatures** | **AverageAccuracy** |
| **2000** | **2** | **FALSE** | **2** | **96.495** |
| 1000 | 2 | FALSE | 2 | 97.9727 |
| 500 | 2 | TRUE | 2 | 97.682 |
| 1500 | 2 | FALSE | 1 | 96.9347 |
| 1000 | 4 | TRUE | 3 | 97.517 |
| 1000 | 2 | FALSE | 2 | 97.9727 |
| 500 | 1 | TRUE | 1 | 96.1882 |
| 1000 | 1 | TRUE | 3 | 96.932 |
|  |  |  |  |  |
|  |  |  |  |  |
| 1000 | 1 | FALSE | 1 | 96.932 |
| 1000 | 5 | TRUE | 5 | 94.365 |
| 2000 | 4 | FALSE | 3 | 96.907 |
| 2000 | 2 | TRUE | 2 | 96.4953 |
| 1000 | 4 | TRUE | 5 | 95.517 |
| 1000 | 4 | FALSE | 4 | 94.517 |
| 2000 | 4 | TRUE | 4 | 96.9064 |

**9)Logistic Regression**

|  |  |
| --- | --- |
| **Parametre**  **Family** | **Average Accuracy** |
| Binomial | 44.62 |
| Quasi Binomial | 44.62 |

**10)Bagging**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Parametre-1**  **Mfinal** | **Parametre-2**  **lengthdivisor** | **Parametre-3**  **Iterations** | **Parametre-4**  **cp** | **AverageAccuracy** |
| **10** | **4** | **1000** | **0** | **96.64** |
| 20 | 4 | 1000 | 0 | 95.79 |
| 20 | 2 | 500 | 1 | 63.25 |
| 5 | 2 | 500 | 1 | 63.25 |
| 30 | 2 | 500 | 1 | 63.25 |
| 30 | 4 | 500 | 0 | 95.79 |
| 30 | 4 | 1000 | 0 | 95.79 |
| 30 | 4 | 1000 | 2 | 63.58 |
| 30 | 2 | 1000 | 3 | 68.25 |
| 20 | 4 | 1000 | 0 | 95.79 |

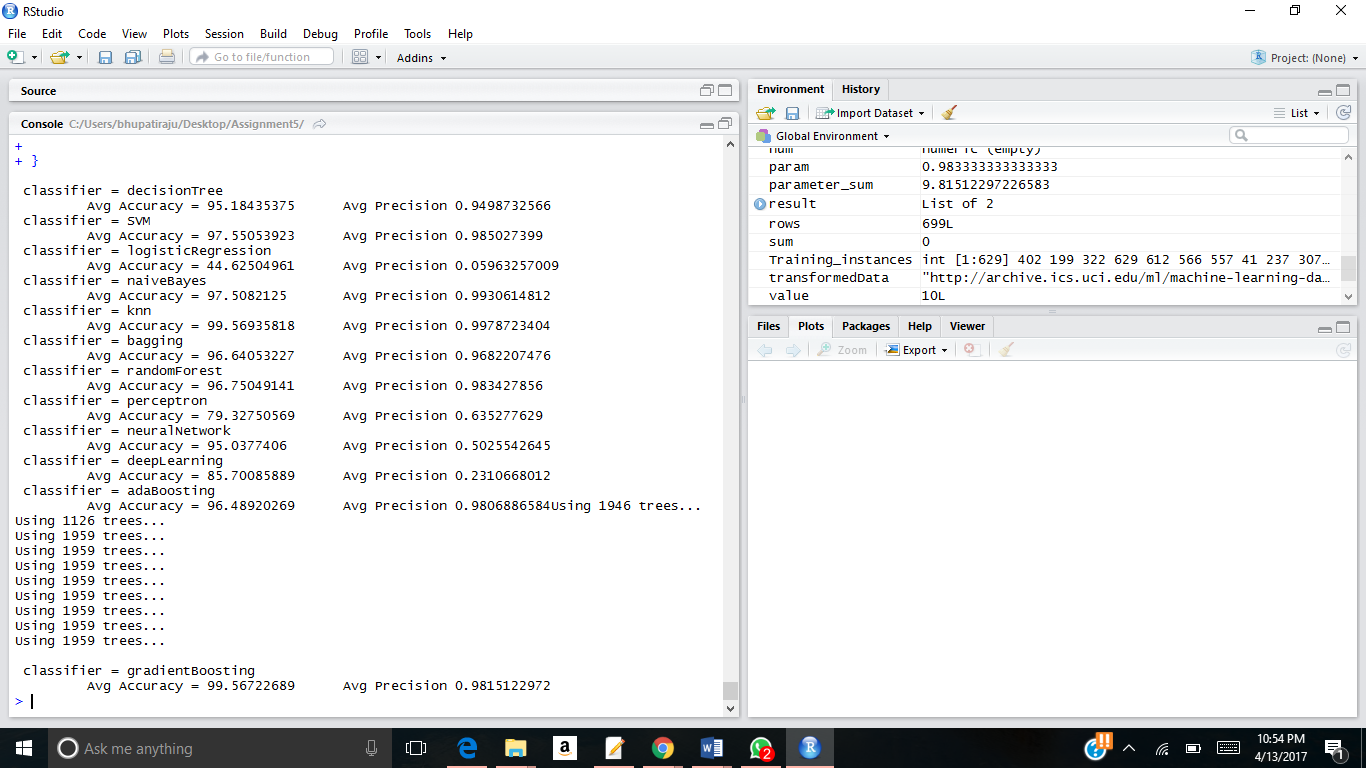
**11)AdaBoosting**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Parametre-1**  **iterations** | **Parametre-2**  **delta** | **Parametre-3**  **bagfraction** | **Parametre-4**  **type** | **Average Accuracy** |
| 20 | 1 | 20 | Real | 86.061 |
| 20 | 1 | 20 | Discreet | 96.1992 |
| **20** | **2** | **20** | **gentle** | **97.21132** |
| 10 | 3 | 10 | gentle | 96.201 |
| 10 | 3 | 20 | gentle | 96.201 |
| 10 | 3 | 20 | Real | 77.74 |
| 10 | 3 | 20 | Discrete | 95.7553 |
| 50 | 2 | 20 | discrete | 97.04 |
| 50 | 2 | 20 | Gentle | 96.4533 |
| 100 | 4 | 50 | gentle | 97.05 |
| 100 | 4 | 50 | Discrete | 96.9040 |
| 50 | 1 | 20 | Discrete | 97.0416 |
| 80 | 4 | 20 | Discrete | 96.8006 |
| 80 | 1 | 20 | Gentle | 96.214 |
| 20 | 4 | 20 | gentle | 97.21132 |

**12)Gradient Boosting**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Parametre-1**  **ntree** | **Parametre-2**  **shrinkage** | **Parametre-3**  **Interaction depth** | **Parametre-4**  **Bag fraction** | **Accuracy** |
| 1200 | 0.01 | 7 | 0.9 | 96.1698 |
| 1000 | 0.01 | 6 | 0.9 | 98.5309 |
| 800 | 0.02 | 7 | 0.8 | 99.275 |
| 800 | 0.01 | 7 | 0.8 | 99.275 |
| 800 | 0.01 | 10 | 0.8 | 99.275 |
| 500 | 0.02 | 10 | 0.6 | 97.469 |
| 400 | 0.01 | 8 | 0.6 | 97.469 |
| **2000** | **0.02** | **10** | **0.4** | **99.56** |
| 600 | 0.02 | 10 | 0.4 | 98.67 |
| 700 | 0.02 | 8 | 0.4 | 98.55 |
| 800 | 0.01 | 8 | 0.7 | 98.384 |
| 2600 | 0.04 | 8 | 0.6 | 98.810 |
| 200 | 0.01 | 10 | 0.7 | 98.550 |
| 3000 | 0.03 | 10 | 0.6 | 98.5507 |
| 500 | 0.03 | 8 | 0.6 | 97.4692 |

**Screenshot**

****

Though KNN had highest accuracy on precision, we cannot directly conclude that, because there is not a significant difference from gradient boosting, NaiveBayes and SVM. Comparatively, a weaker method on the taken dataset was Logistic regression which resulted in low accuracy as well as preciseness. May be because it predicts the outcomes based on independent attributes and is also prone to overfit the training instances which did not suite the particular data set. So the proper selection of algorithm at relevant scenarios is of high importance in machine learning to provide significant results.