1 Overview of Dataset

1.1 Introduction

The dataset I pick to explore is "Students' Academic Performance Dataset", available on Kaggle. The URL is https://www.kaggle.com/aljarah/xAPI-Edu-Data.

The related article is Amrieh, E. A., Hamtini, T., & Aljarah, I. (2016). Mining Educational Data to Predict Student's academic Performance using Ensemble Methods. International Journal of Database Theory and Application, 9(8), 119-136.

It can be downloaded here:

https://www.researchgate.net/publication/307968552_Mining_Educational_Data_to_Predict_Student's_academic_Performance_using_Ensemble_Methods

This is an educational data set which is collected from learning management system (LMS) called Kalboard 360. This system provides users with a synchronous access to educational resources from any device on the Internet.

The dataset is a single csv file of 16 attributes (features) and one class attribute. The total number of instances in the dataset is 480, without missing values. The dataset is also uploaded to Moodle accompanied with this narrative document.

The 16 features are classified into three major categories:

(1) Demographic features such as gender and nationality.
(2) Academic background features such as educational stage, grade level and section.
(3) Behavioral features such as raised hand on class, opening resources, answering survey by parents, and school satisfaction.

The dataset consists of 305 males and 175 females. The students come from different countries.

The dataset is collected through two educational semesters: 245 student records are collected during the first semester and 235 student records are collected during the second semester.

The dataset includes the school attendance feature such as the students are classified into two categories based on their absence days: 191 students exceed 7 absence days and 289 students with absence days under 7.

This dataset also includes a new category of features. It is parent parturition in the educational process. Parent participation feature has two sub features: Parent Answering Survey and Parent School Satisfaction. There are 270 of the parents answered survey and 210 are not, 292 of the parents are satisfied from the school and 188 are not.

1.2 Attributes

1 Gender - student's gender (nominal: 'Male' or 'Female')

2 Nationality- student's nationality (nominal:' Kuwait',' Lebanon',' Egypt',' SaudiArabia',' USA',' Jordan',' Venezuela',' Iran',' Tunis',' Morocco',' Syria',' Palestine',' Iraq',' Lybia')

3 Place of birth- student's Place of birth (nominal:' Kuwait',' Lebanon',' Egypt',' SaudiArabia',' USA',' Jordan',' Venezuela',' Iran',' Tunis',' Morocco',' Syria',' Palestine',' Iraq',' Lybia')

4 Educational Stages- educational level student belongs (nominal: 'lowerlevel','MiddleSchool','HighSchool')

5 Grade Levels- grade student belongs (nominal: ‘G-01’, ‘G-02’, ‘G-03’, ‘G-04’, ‘G-05’, ‘G-06’, ‘G-07’, ‘G-08’, ‘G-09’, ‘G-10’, ‘G-11’, ‘G-12 ‘)

6 Section ID- classroom student belongs (nominal:’A’,’B’,’C’)

7 Topic- course topic (nominal:’ English’,’ Spanish’, ‘French’,’ Arabic’,’ IT’,’ Math’,’ Chemistry’, ‘Biology’, ‘Science’,’ History’,’ Quran’,’ Geology’)

8 Semester- school year semester (nominal:’ First’,’ Second’)

9 Parent responsible for student (nominal:’mom’,’father’)

10 Raised hand- how many times the student raises his/her hand on classroom (numeric:0-100)

11- Visited resources- how many times the student visits a course content(numeric:0-100)

12 Viewing announcements-how many times the student checks the new announcements(numeric:0-100)

13 Discussion groups- how many times the student participate on discussion groups (numeric:0-100)

14 Parent Answering Survey- parent answered the surveys which are provided from school or not (nominal:’Yes’,’No’)

15 Parent School Satisfaction- the Degree of parent satisfaction from school(nominal:’Yes’,’No’)

16 Student Absence Days-the number of absence days for each student (nominal: above-7, under-7)

1.3 Class Label

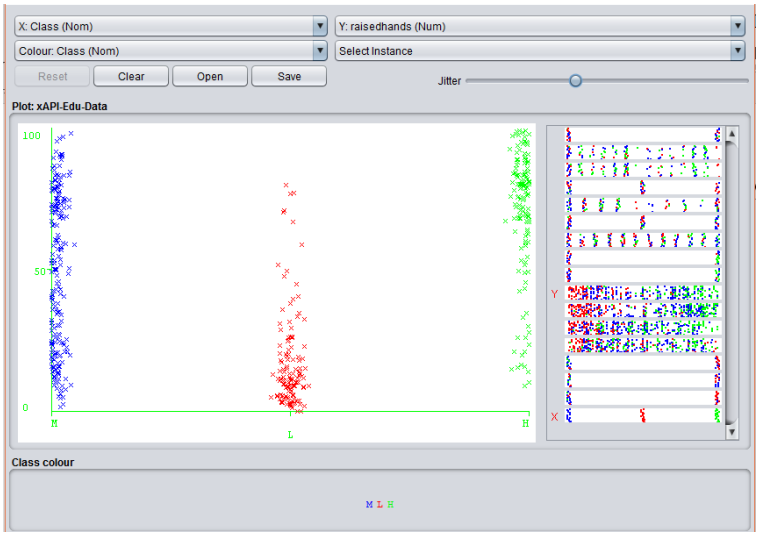
The students are classified into three numerical intervals based on their total grade/mark:

- Low-Level: interval includes values from 0 to 69,
- Middle-Level: interval includes values from 70 to 89,
- High-Level: interval includes values from 90-100.

2 Visualization of Dataset

2.1 Times of students raising hands vs Class

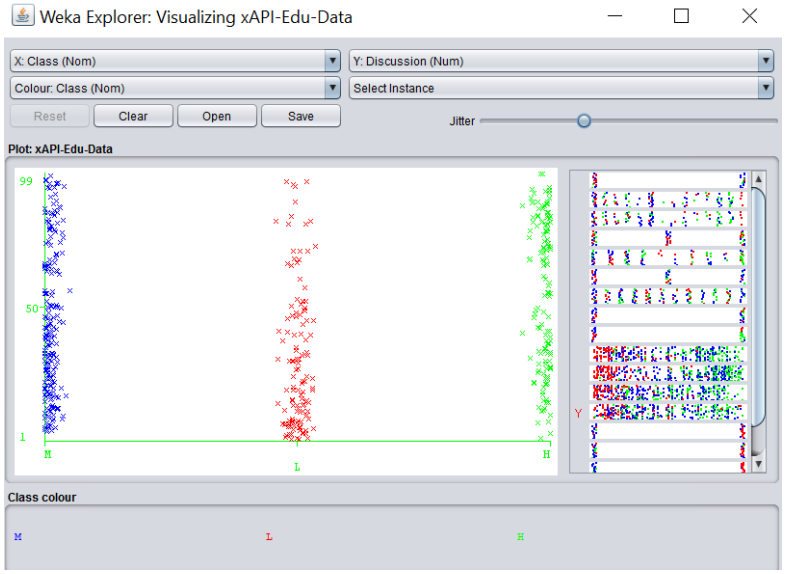
Click visualize tab in Weka, and set x and y axis separately. This is the result:



It can be easily found that generally students with low level of academic performances seldom raise their hands in class. Students with high level of performances raise their hands much more than students with low level of performance. The number of times of raising hands is evenly distributed for students with middle level of performance.

2.2 Times of participating discussion groups vs Class

Likewise, I decide to take a look at whether there exists apparent relationship between how many times the student participate on discussion groups versus the class attribute the student belongs to. The result is shown below.

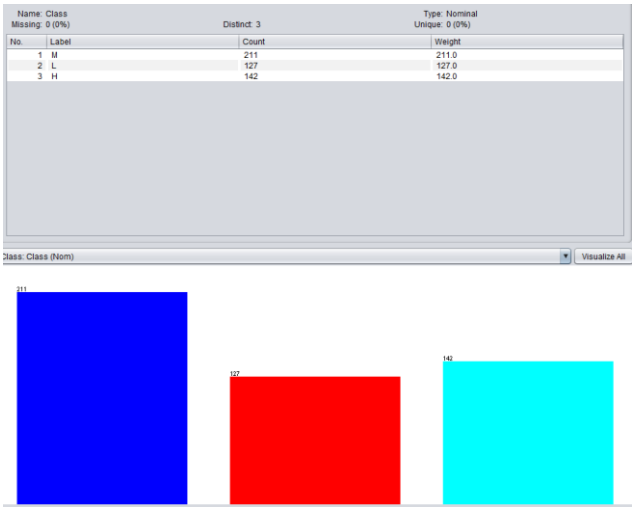


As can be seen above, there is basically no difference between students with middle level and high level of academic performances. They are both evenly distributed in general. While, for students with low level of performance, small amount of them participate in discussions very often.

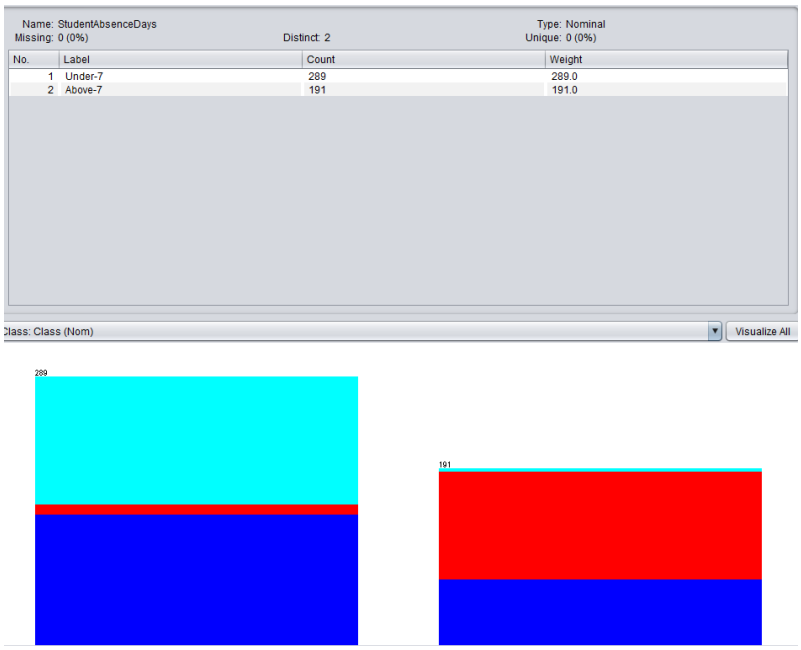
Therefore, compared with raising hands in class, participating in discussion groups is not much related with the performances of the students. Raising hands more often will be more likely to get better grades for students.

2.3 Amount Distribution of Some Attributes

The following is the distribution based on Class attribute.

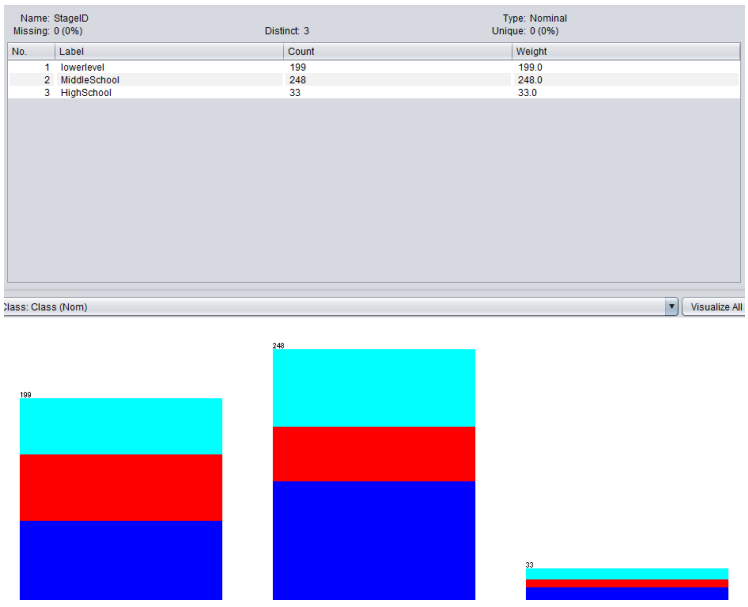


The following is the distribution of absence days.

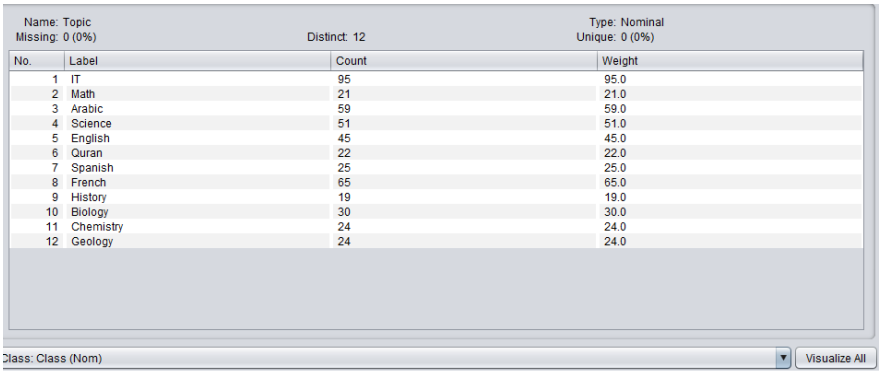


As can be seen above, most of students of low level of performances are absent for school more than 7 days. Most students of high level of performances are absent for school less than 7 days. It implies that less number of absence days for school means better academic performances.

The following is the distribution of StageID. Most students are in lower-level and middle school, only 33 students are from high school. For the first two types of students, we can see that more than one-third students are in middle level of academic performances, which is normal in real life.



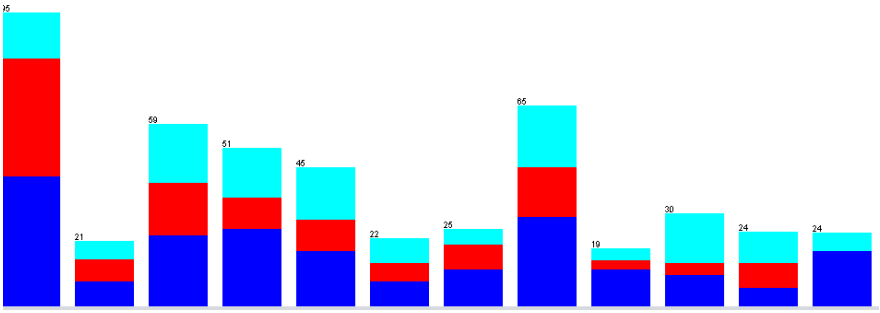
The following is the distribution of topics. IT classes are the most, and History classes are the least. As can be seen, for nearly all the topics, students with high level of academic performances account for less than one-third of the whole. For Geology classes, there is no student with low level of performances.



Name: Topic
Missing: 0 (0%)
Distinct: 12
Type: Nominal
Unique: 0 (0%)

No.	Label	Count	Weight
1	IT	95	95.0
2	Math	21	21.0
3	Arabic	59	59.0
4	Science	51	51.0
5	English	45	45.0
6	Quran	22	22.0
7	Spanish	25	25.0
8	French	65	65.0
9	History	19	19.0
10	Biology	30	30.0
11	Chemistry	24	24.0
12	Geology	24	24.0

Class: Class (Nom) Visualize All

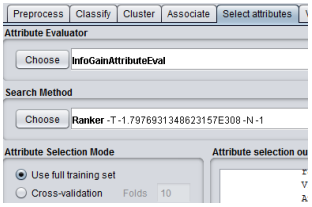


3 Data Cleaning

There is no missing value in the dataset. Therefore, no specific data cleaning is needed. There is no outlier since they are not numeric only values of attributes.

4 Feature (Attribute) Selection

For this section, I use Select attributes tab in Weka. And I choose Information Gain method as evaluator. The result is shown below.




```
Ranked attributes:
 0.45801   11 VisITedResources
 0.39745   16 StudentAbsenceDays
 0.37337   10 raisedhands
 0.2578    12 AnnouncementsView
 0.1504    14 ParentAnsweringSurvey
 0.12773    2 NationalITy
 0.1261     9 Relation
 0.12292    3 PlaceofBirth
 0.11393   13 Discussion
 0.10676   15 ParentschoolSatisfaction
 0.07611    7 Topic
 0.05178    1 gender
 0.04748    5 GradeID
 0.01182    8 Semester
 0.01058    4 StageID
 0.00703    6 SectionID
```

As is shown above, visited resources feature got the higher rank, then followed by student absence days, raised the hand on classroom, parent answering survey, nationality, parent responsible for student, place of birth, discussion groups and parent school satisfaction features. The appropriate subset of features should consist of the top ten features while others can be excluded.

Then, I remove the other lower-ranked features (attributes) in Weka preprocessing tab, only keeping the top 10 attributes.

5 Classification using different models

I use 10-fold cross validation for all the classifiers.

5.1 Naïve Bayes

The result is shown below. The accuracy is 67.9167%. The RMSE is 0.3947. The weighted average precision is 0.680, and the recall is 0.679, the F-measure is 0.671.

```
Correctly Classified Instances         326               67.9167 %
Incorrectly Classified Instances       154               32.0833 %
Kappa statistic                          0.5185
Mean absolute error                      0.2245
Root mean squared error                  0.3947
Relative absolute error                 51.8615 %
Root relative squared error             84.8417 %
Total Number of Instances              480

=== Detailed Accuracy By Class ===

               TP Rate   FP Rate   Precision   Recall  F-Measure  MCC      ROC Area  PRC Area  Class
                 0.521    0.186     0.688      0.521     0.593      0.353    0.774     0.689     M
                 0.858    0.130     0.703      0.858     0.773      0.687    0.951     0.875     L
                 0.754    0.172     0.648      0.754     0.697      0.559    0.876     0.730     H
Weighted Avg.    0.679    0.167     0.680      0.679     0.671      0.502    0.851     0.751

=== Confusion Matrix ===

   a   b   c   <-- classified as
 110  43  58 |   a = M
  18 109   0 |   b = L
  32   3 107 |   c = H
```

5.2 Decision Trees J48

The result is shown below. The accuracy is 72.0833%. The RMSE is 0.3897. The weighted average precision is 0.722, and the recall is 0.721, the F-measure is 0.721.

```
Correctly Classified Instances         346               72.0833 %
Incorrectly Classified Instances       134               27.9167 %
Kappa statistic                          0.5701
Mean absolute error                      0.235
Root mean squared error                  0.3897
Relative absolute error                 54.2842 %
Root relative squared error             83.7644 %
Total Number of Instances              480

=== Detailed Accuracy By Class ===

               TP Rate   FP Rate   Precision   Recall  F-Measure  MCC      ROC Area  PRC Area  Class
                 0.682    0.242     0.689      0.682     0.686      0.441    0.729     0.618     M
                 0.764    0.065     0.808      0.764     0.785      0.712    0.897     0.730     L
                 0.739    0.136     0.695      0.739     0.717      0.593    0.840     0.657     H
Weighted Avg.    0.721    0.164     0.722      0.721     0.721      0.558    0.806     0.659

=== Confusion Matrix ===

   a   b   c   <-- classified as
 144  22  45 |   a = M
  29  97   1 |   b = L
  36   1 105 |   c = H
```

5.3 Logistic Regression

The result is shown below. The accuracy is 72.2917%. The RMSE is 0.3605. The weighted average precision is 0.722, and the recall is 0.723, the F-measure is 0.723.

```
Correctly Classified Instances         347               72.2917 %
Incorrectly Classified Instances       133               27.7083 %
Kappa statistic                          0.5743
Mean absolute error                      0.2287
Root mean squared error                  0.3605
Relative absolute error                 52.8365 %
Root relative squared error             77.4992 %
Total Number of Instances              480

=== Detailed Accuracy By Class ===

               TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
               0.678    0.234    0.694      0.678    0.686      0.445    0.787     0.703     M
               0.803    0.079    0.785      0.803    0.794      0.718    0.914     0.786     L
               0.718    0.124    0.708      0.718    0.713      0.592    0.891     0.741     H
Weighted Avg.  0.723    0.161    0.722      0.723    0.723      0.561    0.851     0.736

=== Confusion Matrix ===

   a   b   c   <-- classified as
 143  27  41 |   a = M
  24 102   1 |   b = L
  39   1 102 |   c = H
```

5.4 Bagging of Decision Trees

The result is shown below. The accuracy is 75.625%. The RMSE is 0.3432. The weighted average precision is 0.757, and the recall is 0.756, the F-measure is 0.756.

```
Correctly Classified Instances         363               75.625  %
Incorrectly Classified Instances       117               24.375  %
Kappa statistic                          0.6226
Mean absolute error                      0.2199
Root mean squared error                  0.3432
Relative absolute error                 50.7895 %
Root relative squared error             73.7699 %
Total Number of Instances              480

=== Detailed Accuracy By Class ===

               TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
               0.749    0.238    0.712      0.749    0.730      0.509    0.815     0.747     M
               0.835    0.059    0.835      0.835    0.835      0.775    0.956     0.854     L
               0.697    0.095    0.756      0.697    0.725      0.617    0.898     0.801     H
Weighted Avg.  0.756    0.148    0.757      0.756    0.756      0.611    0.877     0.791

=== Confusion Matrix ===

   a   b   c   <-- classified as
 158  21  32 |   a = M
  21 106   0 |   b = L
  43   0  99 |   c = H
```

5.5 Bagging of Logistic Regression

The result is shown below. The accuracy is 70.4167%. The RMSE is 0.362. The weighted average precision is 0.706, and the recall is 0.704, the F-measure is 0.705.

```
Correctly Classified Instances         338               70.4167 %
Incorrectly Classified Instances       142               29.5833 %
Kappa statistic                          0.5437
Mean absolute error                      0.2373
Root mean squared error                  0.362
Relative absolute error                 54.8166 %
Root relative squared error             77.8207 %
Total Number of Instances              480

=== Detailed Accuracy By Class ===

               TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
               0.673    0.264    0.667      0.673    0.670      0.409    0.782     0.714     M
               0.787    0.062    0.820      0.787    0.803      0.735    0.940     0.869     L
               0.676    0.145    0.662      0.676    0.669      0.528    0.883     0.722     H
Weighted Avg.  0.704    0.175    0.706      0.704    0.705      0.530    0.854     0.757

=== Confusion Matrix ===

   a   b   c   <-- classified as
 142  21  48 |   a = M
  26 100   1 |   b = L
  45   1  96 |   c = H
```

5.6 Boosting (AdaBoost) of Decision Trees

The result is shown below. The accuracy is 73.9583%. The RMSE is 0.3939. The weighted average precision is 0.740, and the recall is 0.740, the F-measure is 0.740.

```
Correctly Classified Instances         355               73.9583 %
Incorrectly Classified Instances       125               26.0417 %
Kappa statistic                          0.5978
Mean absolute error                      0.1727
Root mean squared error                  0.3939
Relative absolute error                 39.8982 %
Root relative squared error             84.6665 %
Total Number of Instances              480

=== Detailed Accuracy By Class ===

               TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
               0.720    0.242    0.700      0.720    0.710      0.477    0.805     0.732     M
               0.835    0.059    0.835      0.835    0.835      0.775    0.925     0.832     L
               0.683    0.115    0.713      0.683    0.698      0.575    0.888     0.786     H
Weighted Avg.  0.740    0.156    0.740      0.740    0.740      0.585    0.861     0.774

=== Confusion Matrix ===

   a   b   c   <-- classified as
 152  21  38 |   a = M
  20 106   1 |   b = L
  45   0  97 |   c = H
```

5.7 Boosting (AdaBoost) of Logistic Regression

The result is shown below. The accuracy is 72.2917%. The RMSE is 0.3801. The weighted average precision is 0.722, and the recall is 0.723, the F-measure is 0.723.

```
Correctly Classified Instances         347               72.2917 %
Incorrectly Classified Instances       133               27.7083 %
Kappa statistic                          0.5743
Mean absolute error                      0.2882
Root mean squared error                  0.3801
Relative absolute error                 66.5724 %
Root relative squared error             81.7115 %
Total Number of Instances              480

=== Detailed Accuracy By Class ===

               TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
               0.678    0.234    0.694      0.678    0.686      0.445    0.730     0.650     M
               0.803    0.079    0.785      0.803    0.794      0.718    0.853     0.683     L
               0.718    0.124    0.708      0.718    0.713      0.592    0.824     0.642     H
Weighted Avg.  0.723    0.161    0.722      0.723    0.723      0.561    0.790     0.656

=== Confusion Matrix ===

   a   b   c   <-- classified as
 143  27  41 |   a = M
  24 102   1 |   b = L
  39   1 102 |   c = H
```

5.8 Random Forest

The result is shown below. The accuracy is 77.5%. The RMSE is 0.3358. The weighted average precision is 0.775, and the recall is 0.775, the F-measure is 0.775.

```
Correctly Classified Instances         372               77.5    %
Incorrectly Classified Instances       108               22.5    %
Kappa statistic                          0.6524
Mean absolute error                      0.2284
Root mean squared error                  0.3358
Relative absolute error                 52.7537 %
Root relative squared error             72.1725 %
Total Number of Instances              480

=== Detailed Accuracy By Class ===

               TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
               0.763    0.212    0.739      0.763    0.751      0.549    0.828     0.779     M
               0.858    0.059    0.838      0.858    0.848      0.793    0.965     0.874     L
               0.718    0.089    0.773      0.718    0.745      0.644    0.913     0.810     H
Weighted Avg.  0.775    0.135    0.775      0.775    0.775      0.642    0.889     0.813

=== Confusion Matrix ===

   a   b   c   <-- classified as
 161  21  29 |   a = M
  17 109   1 |   b = L
  40   0 102 |   c = H
```

6 Evaluation

All the results are shown in the following table.

Evaluation Measure	Traditional Classification Methods			Bagging		Boosting		Random Forest
Classifier Type	Naïve Bayes	Decision Trees	Logistic Regression	**Decision Trees**	Logistic Regression	Decision Trees	Logistic Regression	**Decision Trees**
Accuracy	67.9167%	72.0833%	72.2917%	**75.625%**	70.4167%	73.9583%	72.2917%	**77.5%**
RMSE	0.3947	0.3897	0.3605	**0.3432**	0.362	0.3939	0.3801	**0.3358**
Precision	0.680	0.722	0.722	**0.757**	0.706	0.740	0.722	**0.775**
Recall	0.679	0.721	0.723	**0.756**	0.704	0.740	0.723	**0.775**
F-Measure	0.671	0.721	0.723	**0.756**	0.705	0.740	0.723	**0.775**

As can be seen above, Random Forest performs best among all the classifiers. Bagging with Decision Trees performs the second best. While, among traditional classifiers, Logistic Regression performs the best. However, no matter combined with Bagging or Boosting, ensemble methods with Logistic Regression performs even worse than itself only. For ensemble methods with Decision Trees, both Bagging and Boosting improve its performances.

In addition to traditional classification methods, ensemble methods are applied to provide an accurate evaluation for the features and to improve the performances of prediction models to some extent. Ensemble methods are categorized into dependent and independent methods. In a dependent method, the output of a learner is used in the creation of the next learner. Boosting is an example of dependent methods. On the contrary, in an independent method, each learner performs independently and their outputs are combined through a voting process. Bagging and random forest are example of independent methods. After applying ensemble methods, individual classifiers results are combined through a voting process, and the class chosen by most number of classifiers is the ensemble decision (Zhou, 2012).

For boosting, I use AdaBoost, adaptive boost. The main idea behind this algorithm is to pay more attention to patterns that are hard to classify. The amount of attention is measured by a weight that is assigned to every subset in the training set. All the subsets are assigned equal weights. In each iteration, the weights of misclassified instances are increased while the weights of truly classified instances are decreased. Then the AdaBoost ensemble combines the learners to generate a strong learner from weaker classifiers through a voting process (Zhou, 2012).

For bagging, its aim is to increase the accuracy of unstable classifiers by creating a composite classifier, then combine the outputs of the learned classifiers into a single prediction. Bagging starts with resampling the original data into different training data sets called bootstraps, and each bootstrap sample size is equal to the size of the original training set. All bootstrap samples will be trained using different classifiers. Individual classifiers results are then combined through majority vote process, the class chosen was by the most number of classifiers is the ensemble decision (Zhou, 2012). Furthermore, bagging works best with high variance models which produce variance generalization behavior with small changes to the training data. Decision Trees algorithm is an example of high variance models. Therefore, **bagging with Decision Trees enhances all the indicators of performances compared to DT only**, consistent with the result in the above table.

Random Forest is a special modification of bagging where the main difference with bagging is the integration of randomized feature selection. Through the decision tree construction process, Random Forest uses random decision trees to select a random subset of features. Randomness is only performed on the feature selection process, but the choice of a split point on the selected features is performed by bagging. The combination between decision tree and bootstrapping makes Random Forest strong enough to overcome the overfitting problem, and to reduce the correlation between trees which provides an accurate prediction (Zhou, 2012).

References

Z. H. Zhou, “Ensemble methods: foundations and algorithms”, CRC Press, (2012).