Application of K-Means Algorithm for Clustering Student's Computer Programming Performance in Automatic Programming Assessment Tool

Anita Qoiriah¹*, Rina Harimurti¹, Andi Iwan Nurhidayat¹, Asmunin¹

¹Jurusan Teknik Informatika Fakultas Teknik, Universitas Negeri Surabaya, Surabaya, Indonesia
*Email: anitaqoiriah@unesa.ac.id

ABSTRACT

Programming is a course that is considered quite difficult for most students. Students are required to have abilities in all processes. Computer programming skills require a lot of practice through lab work assignments. Managing and assessing results of a student's lab work assignment is complex and quite time-consuming task. The availability of automatic programming assessment tools to receive the results of lab work assignments and automatically correct and assess can ease the task of a lecturer. Grouping students according to their level of performance, makes it easy for lecturers to monitor student performance levels and can provide learning according to students' abilities. Grouping was done using K-Means clustering method. Data was score obtained from the lab work assignment of the Automatic Programming Assessment Tool. From the results of clustering, there were 3 groups of students based on their abilities, namely 16 people in the medium ability group, 11 people were students with high ability and 14 people were students whose programming abilities were still lacking.

Keywords: programming assessment tool, programming, clustering, K-Means

1. INTRODUCTION

Programming is a fundamental science that students taking degree in computer sciences must study. A significant skill must be possessed by student of the IT department to evaluate and to execute programming languages.

Programming is a course that most students considered rather hard. Programming is an aptitude to level learners before they reach greater concentrations to comprehend the basic abilities. For instance, students must gradually learn syntax and then semantics structure.

On the other hand, Programming requires some skills. The programming method begins by converting the issue into an algorithm and translating the algorithm into a program code. The hardest part is the application of the algorithm specifications. The right algorithm will produce a program that is as expected. Consequently, students are required to have the ability in all processes; analyze problems, design algorithms, translate algorithms into program code and write program code with the correct syntax. [7].

Computer programming skills require a lot of practice through lab work assignments. Managing and assessing results of student's lab work assignments is very complex and time-consuming. The availability of automatic programming assessment tools to receive grade of lab work assignments, correcting and giving an assessment automatically can ease the task of a lecturer to evaluate student's lab work assignments. The integration of such tools into the Learning Management System is an important feature to improve its performance.

Performance evaluation is one of the basics to monitor the progress of student performance. Grouping students according to their level of performance makes it easy for lecturers to monitor student performance levels and can provide learning according to the abilities of students in these groups. Each group can have different treatment according to what is needed. Grouping can be done by clustering based on the similarity of the performance of each student.
With this grouping, the assignment of practical work and monitoring student performance can be more targeted. In the traditional grouping, students are grouped based on their average scores. In this way, it is difficult to get a comprehensive view of the state of student performance. [6]

The grouping was performed using the K-Means clustering technique through data mining analysis. In this procedure the information was divided into multiple clusters on the basis that the data were similar, so that data with same features were divided into single cluster, while distinct features grouped into another cluster with same features. [5]

This research discussed how the results of the assessment of the automatic programming assessment tool used to classify students based on their performance using the K-Means method.

Automatic Programming Assessment Tools is a tool used to examine programming assignments made by a student. This tool must be able to display lab work assignment, live coding, upload program code, evaluate the program code that has been uploaded, compile, display the compilation results, and do the grading. Besides, this tool can also display scores and rankings from all lab work assignment participants who have uploaded the program code.

In Rina's research, the assessment features developed in the Automatic Programming Assessment Tools referred to the weight of the questions and the correct number of results matching the answer key. The weight of the problem was directly proportional to the level of difficulty and complexity. The more the number of answer keys in the form of input and output, the smaller the percentage of values. [1]

M. Novák's research proposed the integration of plagiarism checking tools namely jPlag and PMD which were used to read source code into a coherent system connected with LMS (Moodle) and IDE. The goal was that sending tasks and checking could be done easily in one system. The system was used in programming courses, object-oriented programming and Java technology [2]. Plagiarism checking was very important, especially in programming tasks.

From the values generated by the automated programming assessment tools, students could be grouped according to their performance. K-means clustering was widely used in educational research. Fan in his research used the K-means algorithm to analyze student data by making improvements to K-means to get better student grouping results. In the proposed algorithm, a method based on network density was used to remove the first outlier. Second, a new method was used to produce the initial cluster center to replace random determinations [9].

Aldi's research used the K-mean method to cluster lecturers' performance whether it was good or not good. The basis used in the assessment used a tri-rhythm basis for the tertiary institution. In assessing performance, data was needed related to student satisfaction with the lecturer. The data used in this study were student satisfaction data. In collecting data using a questionnaire from the Quality Assurance Agency. Variable used were (1) lecturer reliability; (2) responsiveness; (3) guarantees; and (4) empathy. The results of the study were used to improve the performance of lecturers in teaching to improve student satisfaction index [4].

2. METHOD

This chapter consisted of three parts. Section 1 described the data used in this study, section 2 on general methodologies includes steps and section 3 on the clustering methods used in this study.

2.1 Dataset Details

The data in this study were collected from 41 first semester students of the Department of Informatics. Students were participants in the Basic Programming course. The data collected was the result of an Automatic Programming Assessment Tools assessment from a basic programming lab work that uses C ++ Language. The data was collected from a score of 10 lab work assignments that must be completed by each student.

Figure 1 is an automatic programming tool and Figure 2 is an example of a lab work assignment made by a lecturer.
2.2 K-Means Clustering

K-Means aimed to minimize the Sum of Squared Error (SSE) between data objects with many k-centroids. K centroids in this study amounted to 3, they were a group with low, medium and high programming abilities. The steps for the K-Means algorithm [8]:

1. From the data set to be clustered many k objects are chosen randomly as the initial centroid, cj.
2. Every object that is not centroid is inserted into the nearest cluster based on the Euclidean distance measure:  \( \| x_i(j) - c_j \|^2 \).
3. Each centroid is updated based on the average of the objects in each cluster
4. Iterate for the second and third steps until all centroids are converging and stable, where all the centroids produced in the current iteration are the same as all the centroids produced previously.

Several distance measurements were used as a measure of similarity of data, one of which is the Euclidean distance. The Sum of Squared Error (SSE) as in Equation 1:

\[
J = \sum_{j=1}^{k} \sum_{i=1}^{n} \| x_i^{(j)} - c_j \|^2 \quad \text{...............(1)}
\]

Where \( J \) is the objective function, \( k \) is the number of clusters that is 3, \( n \) is the number of data objects that is 41, \( x_i \) is the value of the data object to \( i \), \( c_j \) is the centroid cluster \( j \), \( \| x_i^{(j)} - c_j \|^2 \) is a distance function.

3. RESULT AND DISCUSSION

3.1 Input Data

The test data used was the value of 10 practical tasks generated by the Automatic programming assessment tool. The practicum participants were 41 people. So that the total data was 410 values. Furthermore, the data would be processed into 3 clusters. The cluster described groups of students according to their performance, namely groups that had less, medium and high abilities. Input data was described in graphical form as shown in Figure 4. The x-axis was student and the y-axis showed the grade of lab work assignments. Each lab work assignment was depicted in a different color.
In this study, the data was grouped into 3 groups with low, medium and high programming abilities. In applying the K-means algorithm, these 3 groups were the value of the number of clusters to be created. As an initial centroid, 10 practicum grades were taken from 3 students randomly. Furthermore, from data not centroid, we looked for the minimum distance from the initial centroid. Suppose that an initial centroid is randomly obtained as follows:

<table>
<thead>
<tr>
<th>Cluster</th>
<th>P1</th>
<th>P2</th>
<th>P3</th>
<th>P4</th>
<th>P5</th>
<th>P6</th>
<th>P7</th>
<th>P8</th>
<th>P9</th>
<th>P10</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>87</td>
<td>72</td>
<td>88</td>
<td>91</td>
<td>91</td>
<td>88</td>
<td>93</td>
<td>88</td>
<td>88</td>
<td>86</td>
</tr>
<tr>
<td>2</td>
<td>88</td>
<td>98</td>
<td>82</td>
<td>82</td>
<td>97</td>
<td>88</td>
<td>88</td>
<td>90</td>
<td>98</td>
<td>92</td>
</tr>
<tr>
<td>3</td>
<td>89</td>
<td>88</td>
<td>96</td>
<td>95</td>
<td>98</td>
<td>90</td>
<td>91</td>
<td>87</td>
<td>88</td>
<td>95</td>
</tr>
</tbody>
</table>

The next process was to update the centroid value based on the average value of the object minimum values. For the first data object, the distance to the initial centroid will be searched, so that it will be obtained as follows:

<table>
<thead>
<tr>
<th>Nr</th>
<th>P1</th>
<th>P2</th>
<th>P3</th>
<th>P4</th>
<th>P5</th>
<th>P6</th>
<th>P7</th>
<th>P8</th>
<th>P9</th>
<th>P10</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>93</td>
<td>97</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>95</td>
<td>90</td>
<td>97</td>
<td>100</td>
<td>98</td>
</tr>
</tbody>
</table>

By using the Euclidean distance formula, we obtained the distance to cluster 1 was 37.34, the distance to cluster 2 was 28.72, the distance to cluster 3 was 20.52. The minimum value obtained was the distance to cluster 3, which was 20.52. So the value in cluster 3 was updated to be the midpoint between the centroid in cluster 3 and the data from the object. The centroid value would change to be as follows:

<table>
<thead>
<tr>
<th>Cluster</th>
<th>P1</th>
<th>P2</th>
<th>P3</th>
<th>P4</th>
<th>P5</th>
<th>P6</th>
<th>P7</th>
<th>P8</th>
<th>P9</th>
<th>P10</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>87</td>
<td>72</td>
<td>88</td>
<td>91</td>
<td>91</td>
<td>88</td>
<td>93</td>
<td>88</td>
<td>88</td>
<td>86</td>
</tr>
<tr>
<td>2</td>
<td>88</td>
<td>98</td>
<td>82</td>
<td>82</td>
<td>97</td>
<td>88</td>
<td>88</td>
<td>90</td>
<td>98</td>
<td>92</td>
</tr>
<tr>
<td>3</td>
<td>91</td>
<td>93</td>
<td>98</td>
<td>97.5</td>
<td>99</td>
<td>93</td>
<td>91</td>
<td>92</td>
<td>94</td>
<td>97</td>
</tr>
</tbody>
</table>

The same process was performed for all subsequent data objects. And the process was repeated until a stable centroid was produced, i.e. the centroid value now and then remains the same. A stable midpoint or centroid value was generated as can be seen as in Table 1.

### Table 1. Centroid Data

<table>
<thead>
<tr>
<th>Attribute</th>
<th>cluster_0</th>
<th>cluster_1</th>
<th>cluster_2</th>
</tr>
</thead>
<tbody>
<tr>
<td>P1</td>
<td>87.375</td>
<td>85.192</td>
<td>87.571</td>
</tr>
<tr>
<td>P2</td>
<td>80.938</td>
<td>86.545</td>
<td>85.500</td>
</tr>
<tr>
<td>P3</td>
<td>80.062</td>
<td>81.184</td>
<td>84.571</td>
</tr>
<tr>
<td>P4</td>
<td>44.312</td>
<td>84.182</td>
<td>84.214</td>
</tr>
<tr>
<td>P5</td>
<td>84.312</td>
<td>95.545</td>
<td>91.929</td>
</tr>
<tr>
<td>P6</td>
<td>88.8</td>
<td>84.636</td>
<td>88.500</td>
</tr>
<tr>
<td>P7</td>
<td>89.438</td>
<td>83.818</td>
<td>88.205</td>
</tr>
<tr>
<td>P8</td>
<td>84.250</td>
<td>85.727</td>
<td>89.571</td>
</tr>
<tr>
<td>P9</td>
<td>85.812</td>
<td>86.364</td>
<td>89.714</td>
</tr>
<tr>
<td>P10</td>
<td>80.750</td>
<td>94</td>
<td>89.357</td>
</tr>
</tbody>
</table>

### 3.3 Clustering

By using this centroid, the data would be grouped into 3 clusters. Clustering was determined based on the minimum distance between objects and centroids. The process was repeated by determining the new centroid based on clustering results. The loop was stopped when the same clustering was obtained as before. Clustering results obtained as shown in Figure 5, where cluster 0
contained 16 items or 39% students with moderate ability, cluster 1 as many as 11 items or 27% students with high ability and cluster 2 as many as 14 people or 34% students with lacking programming ability. By looking at the percentage of each group, it appeared that the amount in each group was almost balanced. The results of grouping data with the K-Means algorithm can be described as shown in Figure 6.

![Figure 5. K-Means Clustering Result](image)

**Figure 5. K-Means Clustering Result**

![Figure 6. K-Means Scatter Plot](image)

**Figure 6. K-Means Scatter Plot**

4. CONCLUSION

The K-Means clustering method could be used to group students in programming courses based on their ability. Data was the grade of lab work assignment from the Automatic Programming Assessment Tools. The results of clustering could classify students into 3 clusters; 16 people were in moderate ability groups, 11 students with high abilities and 14 students with programming abilities still lacking.

REFERENCES


