Chen, W. et al. (Eds.) (2017). Proceedings of the 25th International Conference on Computers in Education. New Zealand: Asia-Pacific Society for Computers in Education

Student Placement Predictor for Programming Class Using Classes Attitude, Psychological Scale, and Code Metrics

Ryosuke ISHIZUE[a*], Kazunori SAKAMOTO[b], Hironori WASHIZAKI [a] & Yoshiaki FUKAZAWA [a]
[a]*Department of Science and Engineering, Waseda University, Japan*
[b] *National Institute of Informatics / JST PRESTO, Japan*
*ishizue@ruri.waseda.jp

Abstract: It is often necessary to divide a class according to students' skill level and motivation to learn. This process is burdensome for teachers because they must prepare, implement, and evaluation a placement examination. This paper tries to predict the placement results via machine learning from some materials without such an examination. The explanatory variables are 1. Psychological Scale, 2. Programming Task, and 3. Student-answered Questionnaire. The participants are university students enrolled in a Java programming class. The target variable is the placement result based on an examination by a teacher of the class. Our classification model with Decision Tree has an F-measure of 0.937. We found that the set of the following explanatory variables can yield the best F-measure (0.937): (1) Class Fan Out Complexity, (2) Practical utility value, (3) Difficulty Level 4 (AOJ), (4) Difficulty Level 3 (AOJ), (5) Interest value, and (6) Never-Give-Up Attitude.

Keywords: machine-learning, programming class, placement, psychological scale

1. Introduction

It is often necessary to divide students into an advanced class and an intermediate class based on skill level, motivation to learn, etc. However, dividing students is burdensome on a teacher because the teacher must prepare, implement, and evaluate the examination (e.g., placement test or questionnaire about requests regarding class level) to assess students' ability. Moreover, when the teacher conducts such questionnaire, its interpretation depends on the teacher individually. It causes problems in a class where two or more teachers are assigned or when the teacher changes. Additionally, there are several other problems. For example, some students only memorize the answers of past examinations, while other students cram for a test in one night. This paper aims to properly place students using a method easier than the traditional time-consuming examination.

We focus on a class for second-year undergraduate students learning to program in Java at Waseda University. In this class, students are divided into an advanced class and an intermediate class about a month after the semester begins. Students complete a placement examination by the teacher. In this paper, we try to substitute the examination with a questionnaire, which asks students about their class attitude and the result of a programming task in class. This information is then used to create a machine-learning model to predict the placement results. The explanatory variables are 1. Psychological Scale, 2. Programming Task, and 3. Student-answered Questionnaire. The classification model has precision, recall, and an F-measure of 0.937. Additionally, we evaluate the effects of the explanatory variables on the placement results.

The contributions of this paper are:

- We investigate factors affecting the placement results: (1) Class Fan Out Complexity, (2) Practical utility value, (3) Difficulty Level 4 (AOJ), (4) Difficulty Level 3 (AOJ), (5) Interest value, and (6) Never-Give-Up Attitude.
- We create a model with Decision Tree which has an F-measure of 0.937 to predict the placement results.

2. Related Work

We used some famous psychological scales as explanatory variables in machine learning. The following scales are thought to affect academic performance. Deci and Ryan (1985, 2002) studied intrinsic motivation in human behavior. They defined intrinsic motivation as the life force or energy for the activity and for the development of the internal structure. The degree of self-efficacy affects the efficiency of that behavior. According to Bandura (1997), self-efficacy expectancies determine the initial decision to perform a behavior, the effort expended, and persistence in the face of adversity. Sherer et al. (1982) developed a self-efficacy scale.

Task value is a scale focusing on the value aspect of motivation. According to Eccles and Wigfield (1985), task value is divided into three subscales (interest value, attainment value, and utility value). Moreover, Ida (2001) divided task further divided attainment value and utility value into two for a total of five subscales. Attainment value is divided into private attainment value, which positions him/herself with absolute standards by individuals, and public attainment value, which focuses on attention to superiority/inferiority with others. Utility value is divided into institutional utility value, which is used when learning is necessary to pass an examination for employment or admission, and practical utility value, which is used when learning is useful in occupational practice. Ida (2001) also proposed a task value evaluation scale.

According to Duckworth, and Quinn (2009), self-control is needed to achieve goals that require long-term effort. Self-control allows one to focus on a goal (Consistency of Interest) and persevere through difficulties (Perseverance of Effort). They called this combination Grit, and developed an evaluation scale.

Goal orientation is divided into three subscales: mastery orientation, performance approach, and performance avoidance. Elliot and Church (1997) examined their influences and factors. Multi-dimensional competitiveness is divided into three subscales: Instrumental Competitiveness, Avoidance of Competition, and Never-Give-Up Attitude. Ryckman, Hammer, Kaczor, & Gold (1990, 1996), Smither & Houston (1992), and Ota (2010) developed a multi-dimensional competitiveness. Specific questions based on these scales are shown in section 3.2.1.

Some studies investigated these psychological scales and learning. For example, Robbins et al. (2004) examined the relationship between psychosocial and study skill factors (PSFs) and college outcomes. They found that the best predictors for grade point average (GPA) are academic self-efficacy and achievement motivation. Shen, Chen, & Guan (2007) investigated the potential influence of mastery goal, performance-approach, and avoidance-approach goals, individual interest, and situational interest on students' learning in a physical education. They reported that a mastery goal is a significant predictor for the recognition of situational interest.

Machine learning has been used in various fields, including education. In this paper, we use classification machine learning. For example, Sohsah, Guzey, and Tarmanini (2016) classified educational materials in low-resource languages with machine learning. Márquez-Vera, et al. (2016) predicted school dropout rates of high school students at different steps in the course to determine the best indicators of dropout.

3. Method

We used machine learning (supervised learning) to predict the student placement results for a Java programming class in Waseda University. Three explanatory variables were employed: 1. Psychological Test, 2. Programming Task, and 3. Class Questionnaire. Then we evaluated an effective algorithm and the explanatory variables. The results were used to create and evaluate a model. We used a Python library called *malss* (https://github.com/canard0328/malss/) for the machine learning.

This paper investigated the following research questions (RQs):

- RQ1: How much does each explanatory variable predict the placement results?
- RQ2: What is the best combination of explanatory variables to predict the placement results?

Table 1: Psychological questions.

#	Statements
1	I like programming.
2	I am good at programming.
3	I feel learning to program is interesting.
4	Programming is necessary for my desired job / advancement examination.
5	Programming is useful for desired job / advancement examination.
6	Programming is necessary for practice in my desired occupation.
7	Programming is useful in my desired occupation.
8	I think that learning to program helps me grow as a person.
9	I think that other people respect those who are proficient at programming.
10	I think that to learn programming can be bragging.
11	Setbacks don’t discourage me.
12	I am diligent.
13	I finish whatever I begin.
14	I am a hard worker.
15	I often set a goal but later choose to pursue a different one.
16	I have difficulty maintaining my focus on projects that take more than a few months to complete.
17	New ideas and projects sometimes distract me from previous ones.
18	I am obsessed with a certain idea or project for a short time but later lose interest.
19	I want to learn to improve my abilities.
20	I want to learn new things and increase my knowledge.
21	I want to learn more so others do not think poorly of me.
22	I want to learn properly so as not to give bad results to those around me.
23	I learn to improve the results of the tests and evaluations compared to the around me.
24	When learning something, I like to earn better grades and higher evaluations than other people.
25	By competing, you can enhance your ability.
26	Competition motivates me.
27	If it is boring, I compete with other people to make it interesting.
28	I do not like to compete.
29	I do not want to compete if possible.
30	I do not want to lose.
31	I feel strongly that I do not want to lose.

3.1 Participants

This study included 65 students. They are second-year undergraduate students at Waseda University in Japan enrolled in a Java programming class. This class is equivalent to the CS1 level. After placement test, 50 students were in the advanced course and 15 were in the intermediate course. In this paper, they solved programming tasks, answered a psychological test, and completed a questionnaire about the class. After the placement, 50 students were in the advanced course and 15 students were in the intermediate course.

3.2 Input Data

Table 2: Psychological scales corresponding to each question.

#question	Psychological scale	Subscale
1	Intrinsic motivation	—
2	Self-efficacy	—
3	Task Values	Interest value
4-5	Task Values	Institutional utility value
6-7	Task Values	Practical utility value
8	Task Values	Private attainment value
9-10	Task Values	Public attainment value
11-14	Grit	Perseverance of Effort
15-18	Grit	Consistency of Interest
19-20	Goal Orientation	Mastery orientation
21-22	Goal Orientation	Performance avoidance
23-24	Goal Orientation	Performance approach
25-27	Multi-dimensional Competitiveness	Instrumental Competitiveness
28-29	Multi-dimensional Competitiveness	Avoidance of Competition
30-31	Multi-dimensional Competitiveness	Never-Give-Up Attitude

We prepared following three materials: 1. Psychological Scale, 2. Programming Task, and 3. Class Questionnaire. Materials 1 to 3 were used as explanatory variables in machine learning.

3.3 Psychological Scales

Participants completed a psychological test. Table 1 shows the questions. Each question was evaluated on a seven-level scale: 1. Strongly Agree, 2. Agree, 3. Somewhat Agree, 4. Neutral, 5. Somewhat Disagree, 6. Disagree, 7. Strongly Disagree.

Table 2 shows the psychological scales corresponding to each question. Question 1 measured intrinsic motivation. Question 2 measured self-efficacy. We used simple typical questions such as “I like ~.”, and “I am good at ~.” Questions 3 to10 were based on the task value scale (Eccles and Wigfield, 1985). We used question statements developed by Ida (2001). Questions 11 to 18 were based on the Short Grit Scale (Duckworth, and Quinn, 2009). We used question statements developed by Nishikawa (2015). Questions 19 to 24 were based on Goal Orientation (Tanaka, and Yamauchi, 2000). Questions 25 to 31 were based on Multi-dimensional Competitiveness (Ota 2010).

3.4 Programming Task

We conducted programming tasks in every class. We used Aizu Online Judge (AOJ) to conduct these programming tasks. AOJ is one of the most famous Online Judging System in Japan. AOJ has many programming problems. There are various problems from simple problems such as "Hello World" to difficult problems such as ACM-ICPC (https://icpc.baylor.edu/) past problems. When a user submits his or her program source code via the submission form available on the problem sheet, AOJ checks the correctness of the program on the server side. Table 3 lists the IDs and names of the problems used. Additionally, we set the difficulty level for each problem by considering the correct answer rate, etc. A larger number indicates a more difficult level. Moreover, we measured the source code metrics, which students submitted to AOJ. To collect their source codes, we used *Nightmare* which is a high-level browser automation library written in JavaScript. To measure the metrics, we used *Checkstyle*, which is a static analysis tool for Java. We could make an automatic measurement program easily which has from 100 to 200 LOC because each library has simple APIs. The maximum values determined by Checkstyle’s default were used to detect if the maximum value was exceeded for the following metrics: 1. *Is Solved*, 2. *LOC*, 3. *Boolean Expression Complexity*, 4. *Class Data Abstraction Coupling*, 5. *Class Fan Out Complexity*, 6. *Cyclomatic Complexity*, 7. *Executable Statement Count*, 8. *Max Len file*, 9. *Max Len method*, 10. *Max Line Len*, 11. *Max Outer Types*, 12. *Max Param*, 13. *NCSS Class*, 14. *NCSS File*, 15. *NCSS Method*, 16. *Npath Complexity*, 17. *Npath Complexity*, 18. *Too Many Methods*.

Table 3: Problem id, name, and difficulty of programming task of AOJ (All problems are available from **http://judge.u-aizu.ac.jp/onlinejudge/description.jsp?lang=en?id=ProblemID**).

Problem ID	Problem Name	Difficulty Level
10000	Hello World	1
10001	X Cubic	1
10002	Rectangle	1
10009	Circle	2
10010	Simple Calculator	3
10003	Small Large or Equal	1
10004	Sorting Three Numbers	1
10005	Print Many Hello World	1
10006	Print Test Cases	1
10012	Print Rectangle	1
10013	Print a Frame	2
10016	Grading	2
10019	Sum of Numbers	2
10017	How many ways?	3
10021	Finding minimum String	3
10028	Sort I	3
0121	Seven Puzzle	4
0030	Sum of Integers	4
10014	Print a Chessboard	1
ITP1_5_D	Structured Program I	1
10023	Shuffle	2
10020	Counting Characters	2
1129	HanafudaShuffle	3
10031	Search II	3
1160	How Many Islands?	4
10026	Standard Deviation	1
10020	Counting Characters	1
0011	Drawing Lots	1
1147	ICPC Score Totalizer Software	2
1129	Hanafuda Shuffle	2
2102	Rummy	3
1173	The Balance of the World	3
1166	Amazing Mazes	3
1144	Curling 2.0	4
1133	Water Tank	4
1302	Twenty Questions	4

3.5 Questionnaire about the Class

We implemented a questionnaire about the class. Participants completed the questionnaire during the class after the placement test. 4 shows the questions. All questions were evaluated on a seven-level scale. These questions were created based on the end-of-term questionnaire that Waseda University for all classes.

3.6 Assignment Test (Programming Quiz)

Table 4: Questionnaire about the class.

#	Questions
1	Are you satisfied with the contents of the class so far?
2	How much time do you spend learning class contents outside of the class hours in a week?
3	Do you try to understand the lesson contents?
4	Do you understand the contents of this class?
5	Do you think that class materials are easy to understand?
6	Do you think that the contents of exercises and homework are difficult?
7	Do you think that the number of tasks and homework is too much?
8	Do you think that teachers give classes by grasping the understanding degree of the students?
9	Are you interested in competitive programming like AOJ and contests?
10	Do you think that this lesson is meaningful?

Table 5: Examination programming quiz.

#	Examination sentence
1	Create a program that computes the sum of natural numbers from 1 to 100 and outputs it to the display. Do not use mathematical formulas.
2	Create a program that calculates the sum of squares from 1 to 100 and outputs it to the display. Do not use mathematical formulas.
3	Create a program to calculate a sequence of numbers (Fibonacci numbers: F(0)=0, F(1)=1, F(n)=F(n-1)+F(n-2)). However, the program should be terminated when F (n) exceeds 10000.
4	Create a program to calculate a sequence of numbers (Trivonachs number: T(0)=0, T(1)=0, T(2)=1, T(n)=T(n-1)+T(n-2)+T(n-3)). However, the program should be terminated when T (n) exceeds 10000.
5	Create a program to generate 1000 Java random numbers with natural numbers between 0 and 100. Display their maximum value, minimum value, and average value.
6	Create a program that displays the number of bills (10,000 yen, 5,000 yen, 1 thousand yen) and coins (500 yen, 100 yen, 50 yen, 10 yen, 5 yen, 1 yen) needed to pay the amount after entering a certain amount on the keyboard. The solution should use the fewest bills or coins possible.
7	Create a game to hit an integer, which is generated randomly. When the user inputs a value smaller than the correct answer, display "it is smaller than the correct answer". When the user inputs a value larger than the correct answer, display "it is larger than the correct answer". The random number is an integer from 0 to 999. If the user does not answer correctly within 10 times, displayed “Game Over”.
8	Please indicate the execution result of the following three programs. e.g. for(char c='A'; c<='Z'; c++) System.out.print(c); System.out.print("\n");

Table 5 shows the examination sentences. The examination result was not used as an explanatory variable for machine learning. However, the result was referred by the teacher for class placement. The quiz time was 90 minutes. Additionally, at this test, teachers asked each student which class he/she wants to be in, advanced class or intermediate class (*Hope Class)*.

3.7 Algorithm Selection

This paper used supervised learning algorithms. Five algorithms were tested to create a better model:

- Support Vector Machine with RBF Kernel (SVM)
- Random Forest (RF)
- Support Vector Machine with Linear Kernel (SVML)
- Logistic Regression (LR)
- Decision Tree (DT)

To evaluate the prediction quality of the model, we implemented **Stratified five-fold cross validation**. First, it divides the data set into five pieces so that each label is divided with the same ratio. One piece is used for testing. The remaining four are used for learning. Cross validation calculates the F-measure with precision and recall, verifying each of five divided data sets as test data five times. We'll use the training data as a test set (a closed test).

3.8 Feature Selection

To improve the model, especially to avoid a high variance, we investigated the influence of each explanatory variable. Ineffective variables were excluded. In the psychological test, we converted the answers to the 31 questions into scores (1 to 7 points). Then we calculated the sum of the scores by 15 subscales. Next, we measured the metrics for all the tasks solved by the students. Scores ranked by magnitude of the metrics were used as explanatory variables for machine learning because the number of explanatory variables is enormous if each metric for each problem is used. Moreover, we added the total number of answers, the number of answers per difficulty level [*Number of Solved Tasks (AOJ)*, and *Difficulty Level 1 to 4 (AOJ)*].

Finally, we tried to create a model that improved the evaluation score. First, we used the explanatory variable with the best F-measure. Then we added the explanatory variable with the next best F-measure. This procedure was repeated until the model did not improve.

4. Results and Discussion

4.1 RQ1: How much does each explanatory variable predicts the placement results?

Table 6 shows the results. s. The explanatory variables of the measured metrics show high F-measures. *Self-efficacy* and *interest value* also shows high F-measures. As we expected, the F-measure of *Hope Class* was high. This means that these explanatory variables predict the placement results. However, other F-measures in the psychological scales are not very good. Especially, *Never-Give-Up Attitude, Perseverance of Effort*, and *Intrinsic motivation* show very low F-measure. *Questions about the class* (Q1-10) show F-measures which are higher than those of psychological scales, lower than those of measured metrics though Q7 (Amount of exercises and homework) shows the lowest F-measure. About the *task value*, the utility values show higher values than the attainment values. From the programming tasks using AOJ, *Number of Solved Tasks (AOJ)* and *Difficulty Level 2 (AOJ)* can predict the placement result to some degree, while *Difficulty Level 4 (AOJ)* shows a low F-measure.

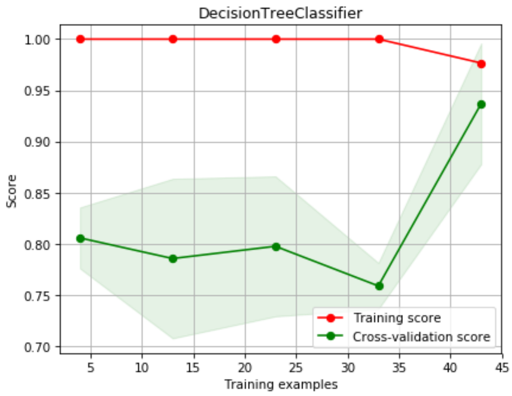
4.2 RQ2: What is the best combination of the explanatory variables to predict the placement results?

We added explanatory variables one by one until the F-measure no longer improved. The best F-measure has a value of 0.937 with DC using the following explanatory variables: (1) *Class Fan Out Complexity*, (2) *Practical utility value*, (3) *Difficulty Level 4 (AOJ)*, (4) *Difficulty Level 3 (AOJ)*, (5) *Interest value*, and (6) *Never-Give-Up Attitude*. Adding more explanatory variables actually decreases the F-measure. A result of a closed test was 0.97. Table 7 shows the F-measures of each algorithm and the best model. By comparing these results, we found that DC was the best algorithm. Figure 1 (right) shows precision, recall, and F-measure of DC. Figure 1 (left) shows a learning curve of DC. Additionally, Figure 2 shows the learning curves of the other algorithms. By comparing these learning curves, there are large gaps between the training scores and cross-validation scores of DC, SVML, RF, and SVM. It means high variance (over-fitting). Thus, if we use more training samples, it can reduce the effect of over-fitting, and lead to improvements in a high variance estimator. On the other hand, even training score of LR is unacceptably bad. It means high bias (under-fitting). Thus, if we use this algorithm and add more features, it can improve a high-bias estimator.

From the result of RQ 1, we expected the results to contain many explanatory variables based on the measured metrics. However, we did not expect *Never-Give-Up Attitude* and *Difficulty Level 4 (AOJ)* to be included because it shows a low F-measure in the previous section.

Table 4: The F-measure of each explanatory variable, the algorithm with the best value, the name of each explanatory variable, and the meaning of each explanatory variable. (Q corresponds to # of Table 4: Questions in the questionnaire about the class. Numbers in parenthesis indicate the maximum values.)

F-measure	Algorithm	Explanatory Variable Name	Meaning
0.700	SVM	Q1	Satisfaction with class
0.669	SVML	Q2	Learning time
0.675	SVM	Q3	Effort to understand the contents
0.762	SVM	Q4	Comprehension of class contents
0.669	SVM	Q5	Ease of understanding class materials
0.670	SVML	Q6	Difficulty of tasks and homework
0.648	SVM	Q7	Amount of exercises and homework
0.669	SVML	Q8	Teacher's understanding of students' level
0.757	SVM	Q9	Interest in competitive programming
0.787	SVM	Q10	Whether the class is meaningful
0.667	Dt	Perseverance of Effort	Long-term efforts to achieve the goals
0.669	RF	Consistency of Interest	Self-control and ability to focus the goal
0.669	Dt	Mastery Orientation	Enhance ability
0.669	SVM	Performance Avoidance	Superior to others
0.669	RF	Performance Approach	Avoid situations where one's incompetence is obvious
0.700	SVM	Instrumental Competitiveness	Achieve another purpose through competition
0.669	SVM	Avoidance of Competition	Avoid competition
0.653	RF	Never-Give-Up Attitude	Do not want to lose
0.834	SVM	Interest Value	Gain fulfillment and satisfaction
0.682	LG	Institutional Utility Value	Must pass the exam for employment or admission
0.681	Dt	Practical Utility Value	Useful for work and study
0.669	LG	Private Attainment Value	Improve oneself on an absolute scale
0.669	RF	Public Attainment Value	Improve oneself on a relative scale
0.871	LG	Self-efficacy	Confidence of one's own ability
0.669	SVM	Intrinsic Motivation	Motivation by curiosity and interest
0.754	RF	AOJ	Total number of questions answered
0.661	SVM	Difficulty Level 1 (AOJ)	# of answers for level 1 problem
0.820	RF	Difficulty Level 2 (AOJ)	# of answers for level 2 problem
0.700	SVM	Difficulty Level 3 (AOJ)	# of answers for level 3 problem
0.669	SVM	Difficulty Level 4 (AOJ)	# of answers for level 4 problem
0.828	Dt	isSolved	Rank of *AOJ*
0.844	SVML	LOC	lines of code
0.846	SVML	Boolean Expression Complexity	# of &&, ||, &, | and ^
0.827	Dt	Class Data Abstractio Coupling	# of instantiations of other classes
0.897	SVML	Class Fan Out Complexity	# of other classes a given class relies on
0.859	SVML	Cyclomatic Complexity	Min # of possible paths in through source
0.864	SVML	Executable Statement Count	# of executable statements
0.88	SVML	Max Len file	# of files exceeding the max Loc (2000)
0.841	SVML	Max Len method	# of methods exceeding the max Loc (150)
0.814	SVML	Max Line Len	# of lines exceeding the max characters (80)
0.868	SVML	Max Outer Types	# of types declared at the outer (or root) level in a file (1)
0.849	SVML	Max Param	# of parameters exceeding Max (7)
0.834	SVML	NCSS Class	# of classes exceeding the Max non-comment lines in the class (1500)
0.865	SVML	NCSS File	# of files exceeding the Max commenting lines in a file including all top level and nested classes (2000)
0.850	SVML	NCSS Method	# of methods exceeding the Max non-comment lines in the class (50)
0.896	SVML	Npath Complexity	# of possible execution paths through a function (method)
0.865	SVML	Too Many Methods	# of methods exceeding the Max methods at all scope levels (100)
0.870	SVM	Hope Class	Class which each student wants to be in.



closed test score

class	precision	recall	F-measure	# of Data
Advance	0.98	0.98	0.98	50
Intermideate	0.93	0.93	0.93	15
avg / total	0.97	0.97	0.97	65

Figure 1. Learning curve (left) and precision, recall, and F-measure of DC whose is 0.937 (right)

Table 7: The F-measure of each algorithm at the best score (5-fold nested cross validation).

Algorithm	F-measure
Support Vector Machine (RBF Kernel)	0.82
Random Forest	0.834
Support Vector Machine (Linear Kernel)	0.787
Logistic Regression	0.785
Decision Tree	0.937

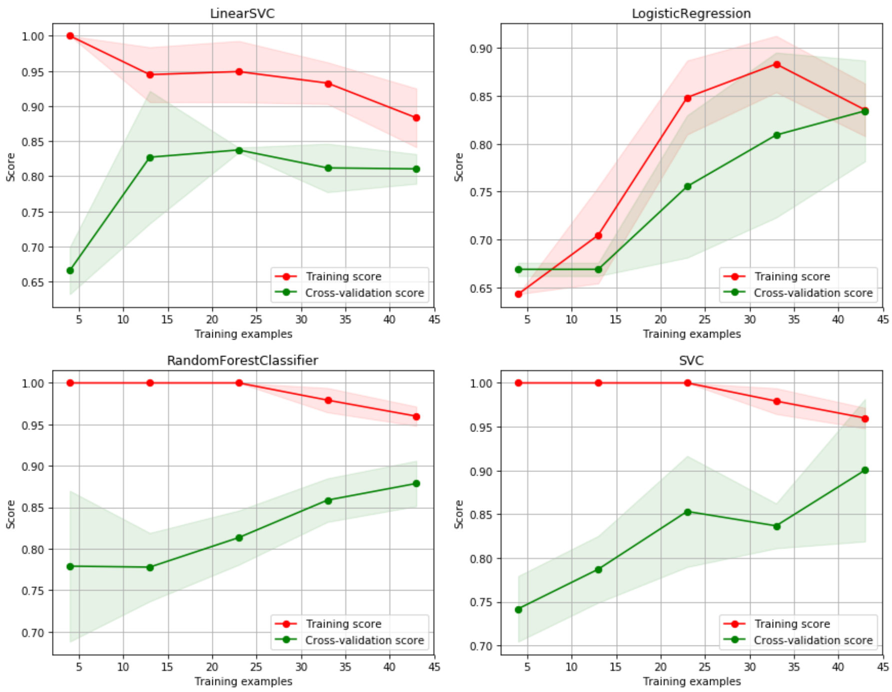


Figure 2. Other learning curves of each algorithm when the F-measure of DC is 0.937.

It is interesting that *Practical utility value* is included. It is thought that these variables performed by combining with the former explanatory variables.

5. Threats to Validity

The questionnaires were conducted after the placement test. This could affect the result. Moreover, the best combination may be a local solution. These are threats to the internal validity.

These results are from one class. If this experiment is repeated with another group or organization, the results may differ. Furthermore, the amount of data is small. These are threats to the external validity.

6. Conclusion and Future Work

Machine learning is used to predict the placement results without a traditional placement examination. The explanatory variables are: Psychological Scale, Programming Task, and Student-answered Questionnaire. The target variable is the Placement Result based on an examination by a teacher. We investigated how these three explanatory variables affect the results. Additionally, we created a classification model with a precision, recall, and F-measure of 0.937.

Additional improvements may be possible. For example, there may be a superior algorithm than the ones used in this study. If our method is expanded in the future, it can be applied other situations such as companies' recruitment and placement.

References

Deci, E. L., & Ryan, R. M. (1985). Intrinsic motivation and self-determination in human behavior. New York: Plenum Press.

Deci, E. L., & Ryan, R. M. (1985). Handbook of self-determination research. Rochester, New York: University of Rochester Press.

Bandura, A. (1977). Self-efficacy: toward a unifying theory of behavioral change. *Psychological Review*, 84, 191-215.

Sherer, M., & Maddux, J. E., Mercandante, B., Prentice-Dunn, S., Jacobs, B., & Rogers, R.W. (1982). The self-efficacy scale: Construction and validation. *Psychological Reports*, 51, 663-671.

Eccles, J. & Wigfield, A. (1985). Teacher expectancies and student motivation. In JB Dusek (Eds.) Teacher expectancies. 185-226. Hillsdale, New Jersey: Erlbaum.

Ida. K. (2001). An Attempt to Construct the Academic Task Values Evaluation Scale. Bulletin of the Graduate School of Education and Human Development. Psychology and human developmental sciences. 49. 83-95.

Robbins, S. B., Lauver, K., Le, H., Davis, D., Langley, R., & Carlstrom, A. (2004). Do psychosocial and study skill factors predict college outcomes? A meta-analysis. Psychological Bulletin, 130, 261-288.

Duckworth, A. L., & Gross, J. J. (2014). Self-control and grit: Related but separable determinants of success. *Current Directions in Psychological Science*, 23, 319-325.

Duckworth, A. L., Peterson, C., Matthews, M. D., & Kelly, D. R. (2007). Grit: Perseverance and passion for long-term goals. *Journal of Personality and Social Psychology*, 92, 1087-1101.

Duckworth, A. L., & Quinn, P. D. (2009). Development and validation of the Short Grit Scale (Grit-S). *Journal of Personality Assessment*, 91, 166-174.

Greenglass, E. R. (2006). Vitality and vigor. In P. Buchwald (Ed.), Stress and anxiety: Application to health, work place, community, and education. Newcastle, UK: Cambridge Scholars Publishing. pp. 65-86.

Nishikawa. K., Okugami. S., and Amemiya. T., (2015). Development of the Japanese Short Grit Scale (Grit-S). *The Japanese journal of personality,* 24, 167-169.

Elliot, A. J., & Church, M. A. 1997 A hierarchical model of approach and avoidance achievement motivation. *Journal of Personality and Social Psychology,* 72, 218-232.

Tanaka, A., & Yamauchi, H. (2000). Causal models of achievement motive, goal orientation, intrinsic interest, and academic achievement in classroom. *The Japanese Journal of Psychology,* 71, 317-324.

Ota, N. (2010). Construction of a Multi-dimensional Competitiveness Scale. *Journal of College of Contemporary Education*, 2, 57-65.

Sohsah, G. N., Guzey, O., & Tarmanini, Z. (2016). Classifying educational materials in low-resource languages. *Proceedings of ICMLA 2016* Anaheim, CA, USA.

Shen, B., Chen, A., & Guan, J. (2007). Using achievement goals and interest to predict learning in physical education. The Journal of Experimental Education, 75(2), 89-108.

Márquez-Vera, C., Cano, A., Romero, C., Noaman, A. Y. M., Mousa Fardoun, H., & Ventura, S. (2016). Early dropout prediction using data mining: a case study with high school students. Expert Systems, 33(1), 107-124.

Kamoshida, R. & Sakamoto, K. (2016). MALSS: A Tool to Support Data Analysis Using Machine Learning for Novices. The Journal of IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences D, 99(4), 428-438.