September Report

Tianze (Steven) Shou

2022-10-21

Introduction

Since the end of the summer, correlation analysis of a set of student-level variables has been carried out. By student-level, we mean that the statistics/data of each student over the three days of experiment have been aggregated, where `student ID` is treated as rows and each data aggreagate act as columns. For short, we can nmae this intermediate data format **student-level aggregate**. The current version of student-level aggreagte data format contains 54 columns that can be categorized into: *observation-related features* (e.g., `totalOnTaskTeacherVisits`), *stop-related features* (e.g., `totalStopLength_rng1000`), *Lynnette-related features* (e.g., `overallPerformance` and `timePerStep`), *detector-related features* (e.g., `idle` and `struggle`), and *test-related features* (e.g., `CPreScore` as conceptual pre-test score and `PLearningGain` as procedural learning gain). A full list of features is below:

`studentID`, `totalOffTaskTeacherVisits`, `totalOnTaskTeacherVisits`, `totalHandRaises`, `totalStopLength_rng500`, `stopsLengthMean_rng500`, `stopsLengthStd_rng500`, `stopsLengthMin_rng500`, `stopsLengthMax_rng500`, `totalStopLength_rng1000`, `stopsLengthMean_rng1000`, `stopsLengthStd_rng1000`, `stopsLengthMin_rng1000`, `stopsLengthMax_rng1000`, `totalStopLength_rng2000`, `stopsLengthMean_rng2000`, `stopsLengthStd_rng2000`, `stopsLengthMin_rng2000`, `stopsLengthMax_rng2000`, `overallPerformance`, `hintRequested`, `problemsSolved`, `totalSteps`, `timePerStep`, `cancel_constPerformance`, `division_simplePerformance`, `dividePerformance`, `subtraction_constPerformance`, `combine_like_constPerformance`, `subtraction_varPerformance`, `combine_like_varPerformance`, `cancel_varPerformance`, `distribute_divisionPerformance`, `division_complexPerform`, `idle`, `struggle`, `gaming`, `misuse`, `PlearningGain`, `ClearningGain`, `IncorrectCount`, `firstAttemptPerformance`, `avgCorrectStepDuration`, `avgErrorStepDuration`, `avgHintDurationPerStep`, `assistanceScorePerStep`, `stopsCountRNG500`, `stopsCountRNG1000`, `stopsCountRNG52000`, `CPreScore`, `PPreScore`, `totalTimeInTutor`, `hintPerStep`, `helpAboveAvg`.

With the student-level aggregate data in-hand, we conducted pair-wise partial correlation analysis for each feature pairs with several variables as control variables. We will further discuss the analysis results in the following sections. For the raw correlation outputs, please refer to the Excel file named `part_corr_tb_annotated.xlsx`, where the correlations and their p-values are all listed and annotated. P-values in yellow cells means that p-value < 0.1, while red signifies p-value < 0.05.

Besides the correlation analysis, we also built some decision tree of to predict students' learning gain and their chances of being helped by the teacher. There will also be corresponding sections for this piece of analysis.

Partial Correlation Analysis

As aforementioned, we calculated all pair-wise correlation for all variables in the student-level aggregate dataset while controlling for *procedural pre-test score*, *conceptual pre-test score*, and *total time spent in tutor*. We eliminated the effect the these features since we do not want the innate capabilities or the amount of opportunities the students have interacting with the tutors to confound our analysis results.

Correlations with *Stop Length*

In this subsection, we are going to discuss other variables' significant correlation with `stop length`. As a brief recap on how `stop length` is being computed, these are the specific steps:

- `Stops` are generated from the three days' position log data with `duration` and `radius` parameters specified. In this case, these parameters are defined as `duration` = 10 secs and `radius` = 0.5 meter.

- `Range` parameter is defined to select students whose positions are within `range` distance from the stop's centroid point. We keep a record of all students within `range` for each stop. In our case, `range` = 0.5 meter.

- All `stops` are parsed, and we keep a mapping where keys are the students and values are the accumulated time length (we call this time length to avoid confusion with parameter `duration`) of the `stops` where the corresponding student appears.

- The accumulated time length would be the eventual aggregated `stop length`.

Below are some variables found significantly correlated with *stop length*, where alpha level is 0.1:
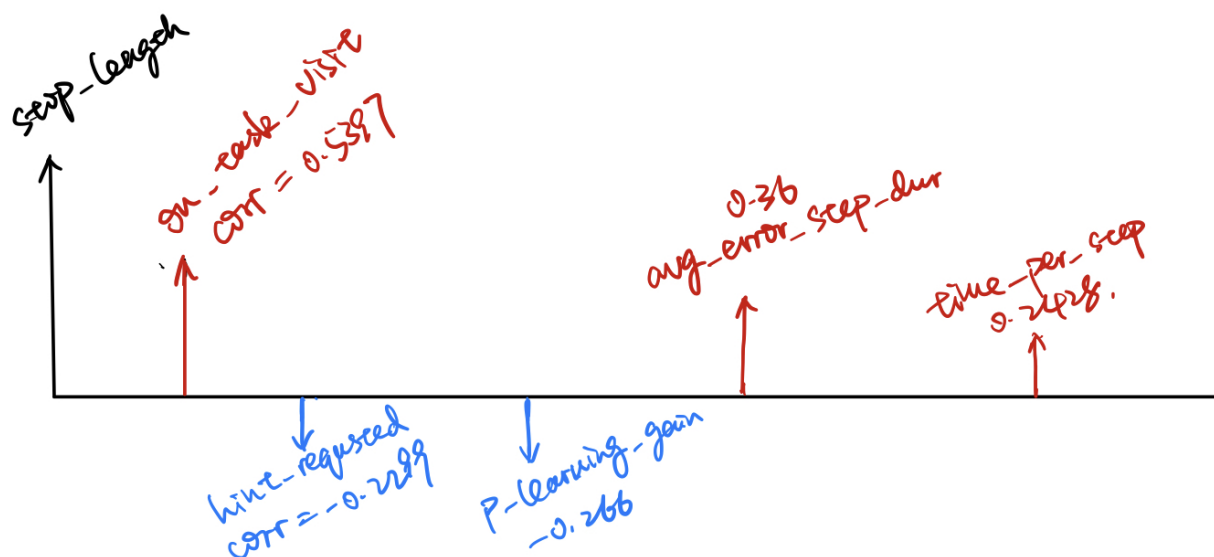

Figure 1: Correlation Visual of Stop Length

In this visual, red arrows signifies positive correlation while blue ones are negative, and the lengths of arrows are also proportional to the magnitude of the correlations. Following is some key interpretations from these correlations, but notice that all relationships do not have sufficient evidence to be shown causal. We can only state that these variables are correlated.

Since the number of on-task visits is significantly, positively correlated with *stop length*, **it means that out step detection algorithm does capture teachers' helping behavior to each student**. We can put more trust into our detection algorithm since it does pass this sanity check. It also seems that **teachers do differentiate students with in-system difficulties** (not using the term *struggle* here since it means something else in detector context) **and pay more attention to them**. From *stop length*'s correlation with *hint requested*, *average error step duration*, and *time per step*, it can be observed that the profile of students who receives more teacher stop/attention would spend more meaningless time trying to figure out the erroneous steps and ordinary steps by themselves and rarely ask for in-system assistance. Although we controlled for the students' *pre-test scores*, *stop length* still demonstrates strong correlation with *procedural learning gain*. This means that the teachers intentionally target their attention to those slow learning students

either by the teachers' preexisting knowledge on the students or by the observations in the current class sessions.

Correlations with *Learning Gain*

On the first and fifth (last) day of the data collection phase, we administered pre and post knowledge tests to all students, where the *conceptual* and *procedural* knowledge levels are assessed. The conceptual questions involve the definition and properties of the math constructs in solving equations, while procedural questions ask "solve for x" type of problems. With pre- and post-test scores calculated as percentiles relative to full marks, we can compute the *learning gain* of each student with the following formula:

$$LG = \frac{post - pre}{1 - pre}$$

One can conceptually understand learning gain as the proportion of progress made by the student in learning process over the maximum progress possible. Since the tests are separated into conceptual and procedural sections, we computed corresponding learning gains separately.

Below are features significantly correlated with *procedural and conceptual learning gains*. Figure 2 and 3 follow the exact same convention as Figure 1.
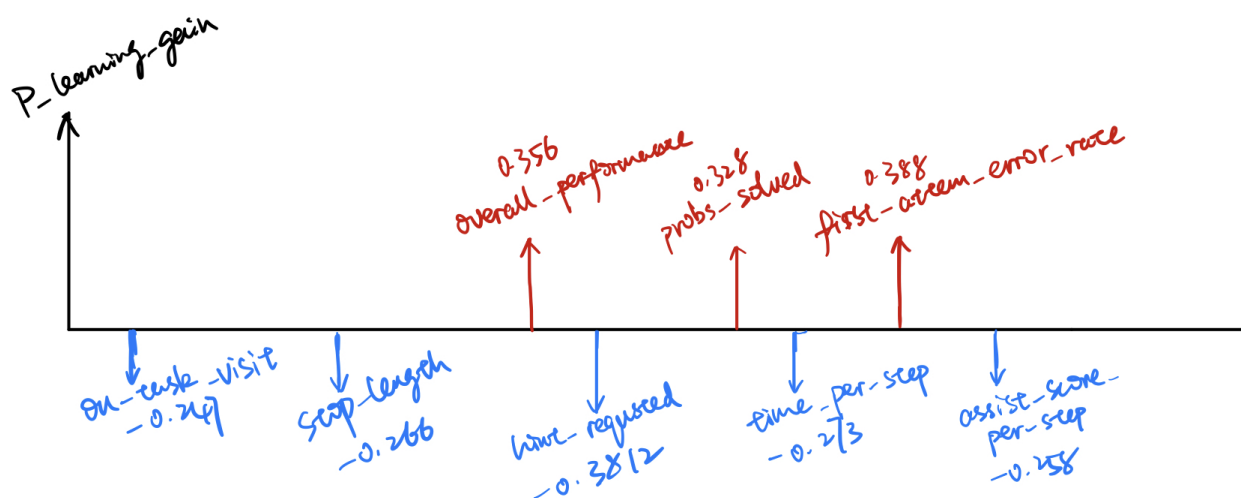

Figure 2: Correlation Visual of Procedural Learning Gain

From Figure 2, *procedural learning gain* is most heavily correlated with teachers' helping behavior (visits, stops) and some in-system performance statistics. **Teachers do tend to assist students moving more slowly in the procedural learning process** since procedural learning gain is negatively correlated with *number of visits* and *stop length*. **Those who learning the procedure more slowly also tend to demonstrate some attributes that are commonly thought as slower learners**. For example, they tend to perform worse overall, ask for more assistance from the tutor, and move more slowly through the problem sets. Moreover, students with better procedural learning gain tend to get more first attempt error, which means that they make more mistakes in the first take of a step. We can take this in two ways: (1) students making first attempt errors have more room for learning gain in similar problems since it is most likely that they have not had any grasp on the material; (2) they gradually grew their knowledge working through the steps they get wrong the first time.

Conversely, *conceptual learning gain* is not significantly related to teachers' helping behavior as an observation from Figure 3, from which we can infer that **teachers' assistance tends to focus more on or is more capable of imparting procedural knowledge than conceptual**. Other in-system statistics' correlations
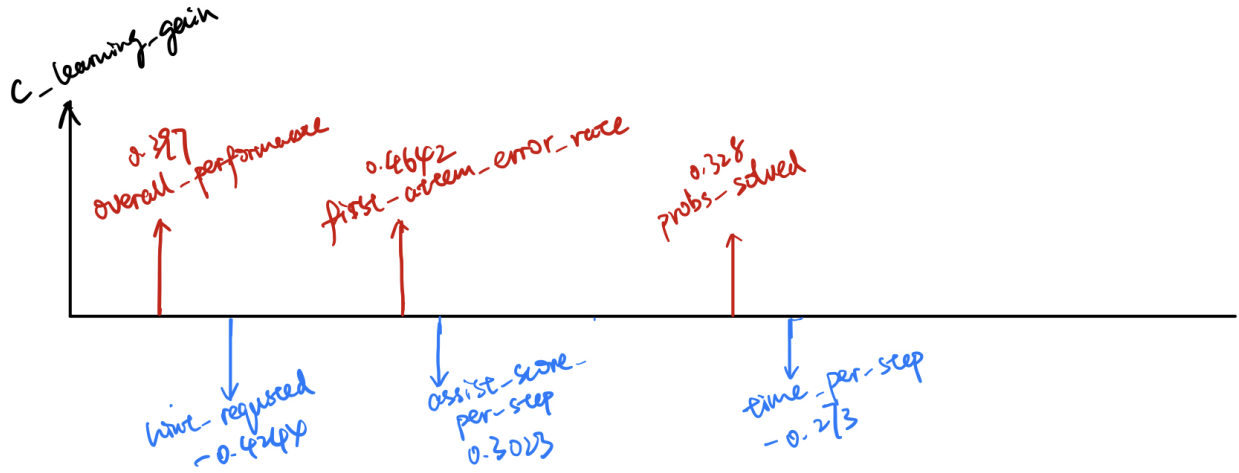
Figure 3: Correlation Visual of Conceptual Learning Gain

closely align with those of *procedural learning gain* in the sense of both magnitude and direction. Slow conceptual learners also demonstrate similar attribute profiles in the tutor system.

Decision Tree Analysis

For this analysis, we separated all students into two groups: students whose *# of teacher visits > mean of teacher visits* and students below mean. We are interested in building a decision tree that help us determine which features are the most important for teachers to determine which student to assist and perhaps simulate teachers' thinking process in choosing students to help. We are going to use the "above mean" and "below mean" measure as target when building the tree.

From Figure 4, we can conclude some key observations. Teachers' helping strategies are very different for those students who raise hands frequently and those who do not actively seek for help. For students who less frequently seek help, which belong to the left subtree from the root node, they usually show some undesired status (e.g., idle, struggle) and prolonged stay in one step and are noticed by the teachers. **Teachers tend to proactively offer assistance to these students since they demonstrate sign of undergoing challenges but do not seek for help**. Those who seek assistance more actively would mostly likely be experiencing more first attempt error. **With little knowledge on how unknown steps can be solved, those student more actively seek help from the teachers**.

Decision tree classifier predicting whether students get on task visits above average number
ABOVE = above average, BELOW = below average

totalHandRaises <= 2.5
gini = 0.49
samples = 84
value = [48, 36]
class = BELOW

timePerStep <= 30.158
gini = 0.451
samples = 67
value = [44, 23]
class = BELOW

firstAttemptPerformance <= 0.775
gini = 0.36
samples = 17
value = [4, 13]
class = ABOVE

gaming <= 567.763
gini = 0.41
samples = 59
value = [42, 17]
class = BELOW

assistanceScorePerStep <= 3.774
gini = 0.375
samples = 8
value = [2, 6]
class = ABOVE

avgErrorStepDuration <= 30.336
gini = 0.375
samples = 4
value = [3, 1]
class = BELOW

problemsSolved <= 60.5
gini = 0.142
samples = 13
value = [1, 12]
class = ABOVE

idle <= 1.307
gini = 0.375
samples = 56
value = [42, 14]
class = BELOW

gini = 0.0
samples = 3
value = [0, 3]
class = ABOVE

gini = 0.0
samples = 6
value = [0, 6]
class = ABOVE

gini = 0.0
samples = 2
value = [2, 0]
class = BELOW

gini = 0.0
samples = 3
value = [3, 0]
class = BELOW

gini = 0.0
samples = 1
value = [0, 1]
class = ABOVE

gini = 0.0
samples = 12
value = [0, 12]
class = ABOVE

gini = 0.0
samples = 1
value = [1, 0]
class = BELOW

totalSteps <= 159.0
gini = 0.439
samples = 43
value = [29, 14]
class = BELOW

gini = 0.0
samples = 13
value = [13, 0]
class = BELOW

IncorrectCount <= 25.0
gini = 0.298
samples = 22
value = [18, 4]
class = BELOW

assistanceScorePerStep <= 0.099
gini = 0.499
samples = 21
value = [11, 10]
class = BELOW

IncorrectCount <= 17.0
gini = 0.48
samples = 10
value = [6, 4]
class = BELOW

gini = 0.0
samples = 12
value = [12, 0]
class = BELOW

gini = 0.0
samples = 5
value = [5, 0]
class = BELOW

avgCorrectStepDuration <= 4.749
gini = 0.469
samples = 16
value = [6, 10]
class = ABOVE

gini = 0.0
samples = 5
value = [5, 0]
class = BELOW

hintRequested <= 78.0
gini = 0.32
samples = 5
value = [1, 4]
class = ABOVE

avgCorrectStepDuration <= 3.61
gini = 0.496
samples = 11
value = [6, 5]
class = BELOW

gini = 0.0
samples = 5
value = [0, 5]
class = ABOVE

gini = 0.0
samples = 4
value = [0, 4]
class = ABOVE

gini = 0.0
samples = 1
value = [1, 0]
class = BELOW

firstAttemptPerformance <= 0.86
gini = 0.408
samples = 7
value = [2, 5]
class = ABOVE

gini = 0.0
samples = 4
value = [4, 0]
class = BELOW

timePerStep <= 15.31
gini = 0.444
samples = 3
value = [2, 1]
class = BELOW

gini = 0.0
samples = 4
value = [0, 4]
class = ABOVE

gini = 0.0
samples = 2
value = [2, 0]
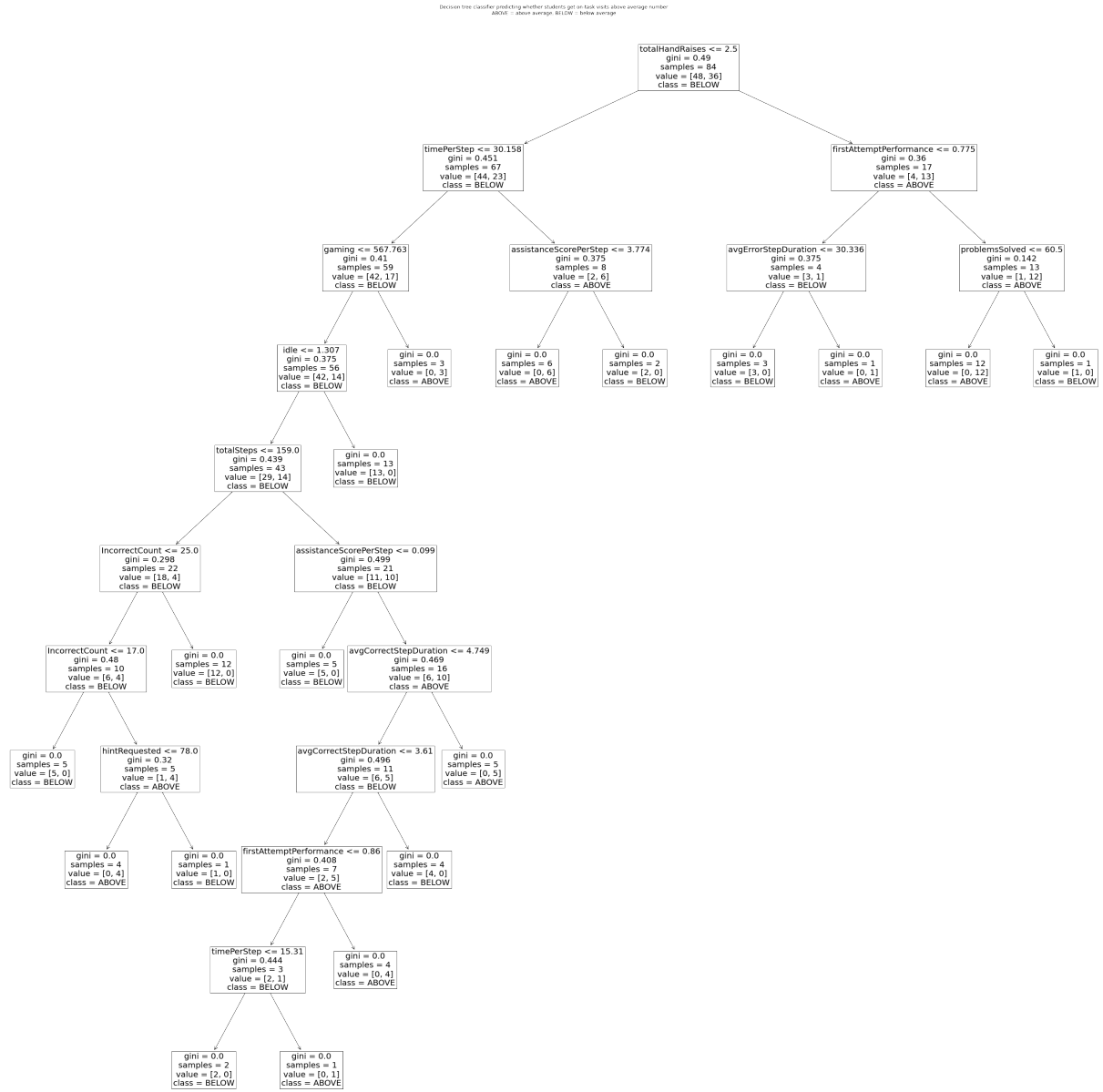class = BELOW

gini = 0.0
samples = 1
value = [0, 1]
class = ABOVE

Figure 4: Decision Tree Predicting Teacher Visits