Journal of Machine Learning Research x (2010) 1-18 Submitted 7/10; Published xx/xx

# Ensemble Hybrid Logit Model

**Poonam Gandhi** POONAM.GANDHI@EXLSERVICE.COM
**Varun Aggarwal** VARUN.AGGARWAL@EXLSERVICE.COM
*EXL Service*
*Gurgaon 122 001, Haryana, India*

**Editor:** xx

## Abstract

This paper summarizes our approach for taking up **KDD Cup 2010 Challenge** [1]. It briefly describes our methodology to predict the future performance of students, based not just on the assessment of their past performance but also on their respective learning curves constructed over time. We present an application of Rasch model technique to capture the effects of student level proficiency and steps' level difficulty. We demonstrate robust validation results from hybrid ensemble of logistic regression models. We also discuss the scope of improved models with segmentation analysis.

**Keywords:** Text Mining, Rasch Model, Random Forest, Segmentation

## 1. Introduction

Let us get back to school days. Let us recall all those old batch-mates. Studying in the *same class*, spending *same number of hours* under the guidance of a *common faculty*, did all of us always score the same grades? The answer, we guess, is **no**. A student's *proficiency* does have a role to play.

If, however, we focus on just one of many, would that student be scoring the same grade after taking the *same* test repeatedly? Again, most likely the answer is **no**. A student's *learning curve* has its own impact. The outcome here is also a function of *difficulty level of each step* in the given problem statement.

Keeping in mind such multiple dimensions, we provide a brief overview of our end-to-end approach in following sections.

## 2. KDD Challenge

### 2.1 Problem Statement as 'Given'

**KDD Cup 2010** challenge is to predict student performance on mathematical problems from logs of student interaction with intelligent tutoring systems. The systems include the *Carnegie Learning Algebra* system and the *Bridge to Algebra* system, deployed 2008-2009.

1. Our KDD Cup 2010 team comprises 27 volunteers (refer Appendix A for complete list) led by **Dr. Krishna Mehta**. We greatly thank him for building up this team and for extending his constant support.

### 2.2 Descriptions of Raw Features

Following are the raw features and their descriptions:

**Row:** This refers to the row number.

**Anon Student Id:** This is unique, anonymous identifier for a student.

**Problem Hierarchy:** This refers to the hierarchy of curriculum levels containing the problem.

**Problem Name:** This is unique identifier for a problem.

**Problem View:** This is total number of times the student encountered the problem so far.

**Step Name:** Each problem consists of one or more steps. The step name is unique within each problem, but there may be collisions between different problems, so the only unique identifier for a step is the pair of problem name and step name.

**Step Start Time:** This refers to the starting time of the step.

**First Transaction Time:** This refers to the time of the first transaction toward the step.

**Correct Transaction Time:** This refers to the time of the correct attempt toward the step, if there was one.

**Step End Time:** This refers to the time of the last transaction toward the step.

**Step Duration:** This refers to the elapsed time of the step in seconds, calculated by adding all of the durations for transactions those were attributed to the step.

**Correct Step Duration:** This refers to the step duration if the first attempt for the step was correct.

**Error Step Duration:** This refers to the step duration if the first attempt for the step was an error (incorrect attempt or hint request).

**Correct First Attempt:** This is the tutor's evaluation of the student's first attempt on the step. It takes value 1 if correct, 0 if an error. This is the *target variable*.

**Incorrects:** This refers to total number of incorrect attempts by the student on the step.

**Hints:** This refers to total number of hints requested by the student for the step.

**Corrects:** This refers to total number of correct attempts by the student for the step. This increases only if the step is encountered more than once.

**KC (KC Model Name):** KC refers to *knowledge component*. A KC model represents a set of identified skills that are used in a problem, where available. A step can have multiple KCs assigned to it. Multiple KCs for a step are separated by two tildes. Since opportunity describes practice by knowledge component, the corresponding opportunities are similarly separated by two tildes.

**Opportunity (KC Model Name):** This refers to a count that increases by one each time the student encounters a step with the listed knowledge component. Steps with multiple KCs have multiple opportunity numbers separated by two tildes.

## 3. Sampling Methodology

Three datasets are being created for both 'algebra' and 'bridge to algebra' problems as a part of sampling methodology.

- **History** dataset for learning and creation of variables at unit, section, problem, student, knowledge component and step level. Variables are created to identify the probability of success at each level by average hit rate in this dataset, average time taken etc. Cross probabilities are also tested (for instance, probability of correct first attempt for a unit when solved by students of particular level). Text mining is used to create several variables using step and problem description. The details of these variables will be discussed in feature creation module.

- **Modeling** dataset for the process of modeling. All the variables created using information contained in history dataset are merged back with this dataset and various modeling techniques like CART and logistic regression are applied on this dataset.

- **Validation** dataset to validate the modeling techniques. This is created in similar fashion as modeling dataset.

The methodology followed for creation of these datasets is exact replication of the methodology used for generation of dataset provided to us. The following steps are followed in both 'algebra' and 'bridge to algebra' problems to generate three datasets:

**Step 1:** For each student and unit, pick all the steps of last attempted problem and insert into *validation dataset*.

**Step 2:** For each student and unit, pick all the steps of second last attempted problem and insert into *modeling dataset*.

**Step 3:** Insert remaining steps for the student and unit into *history dataset*.

Figure 1 summarizes the sampling methodology used.

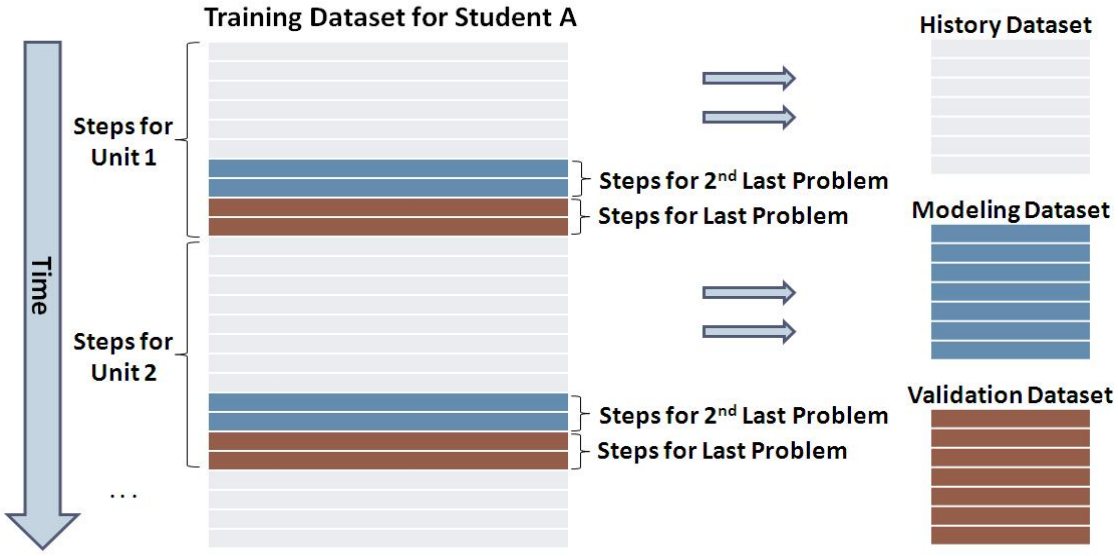


Figure 1: A diagram representing sampling methodology

## 4. Advanced Features Creation

This section describes creation of derived variables.

### 4.1 Knowledge Component Models

**Step 1: Raw Dataset (selected variables)**

| Row | KC (Subskills) | Opportunity (Subskills) |
|---|---|---|
| 1 | XX~~YY~~ZZ~AA | 20~~40~~50~~60 |
| 2 | AA~~XX | 70~~20 |
| 3 | XX~~YY~~ZZ | 40~~30~~70 |
| 4 | PP | 80 |
| . | . | . |
| . | . | . |
| N | . | . |

**Step 3: Create Look Up Table**

| KC_SS | KC_SS_CD |
|---|---|
| | 1 |
| AA | 2 |
| PP | 3 |
| XX | 4 |
| YY | 5 |
| ZZ | 6 |
| . | . |
| . | . |
| . | 634 |

**Step 2: Remove "~~" and Create separate variables**

Note: Here SS refers to SUBSKILLS

| Row | KC_SS_1 | KC_SS_2 | KC_SS_3 | KC_SS_4 | Opp_SS_1 | Opp_SS_2 | Opp_SS_3 | Opp_SS_4 |
|---|---|---|---|---|---|---|---|---|
| 1 | XX | YY | ZZ | AA | 20 | 40 | 50 | 60 |
| 2 | AA | XX | | | 70 | 20 | | |
| 3 | XX | YY | ZZ | | 40 | 30 | 70 | |
| 4 | PP | | | | 80 | | | |
| . | . | . | . | . | . | . | . | . |
| . | . | . | . | . | . | . | . | . |
| N | . | . | . | . | . | . | . | . |

Figure 2: Creation of KC variables look-up table

Figure 2 exhibits a step-by-step process. In Step 1, the dataset is read. A lot of information is being summarized into raw variables like *KC (subskills)* and *opportunity (subskills)*. Such variables need to be split up for use in modeling exercise. In Step 2, therefore, the delimiter (double tilde) is removed and separate variables are created. To use the given information, there is a need to create an indicator for each skill. The actual data does not contain skills like 'AA' and 'XX' but long text strings. Consequently, this calls for assigning a unique code to each knowledge component skill value. With this objective, a look-up table is being created in Step 3. For the purpose of implementation, this look-up table comprises an exhaustive list of all possible values of knowledge components present in both train and test datasets.

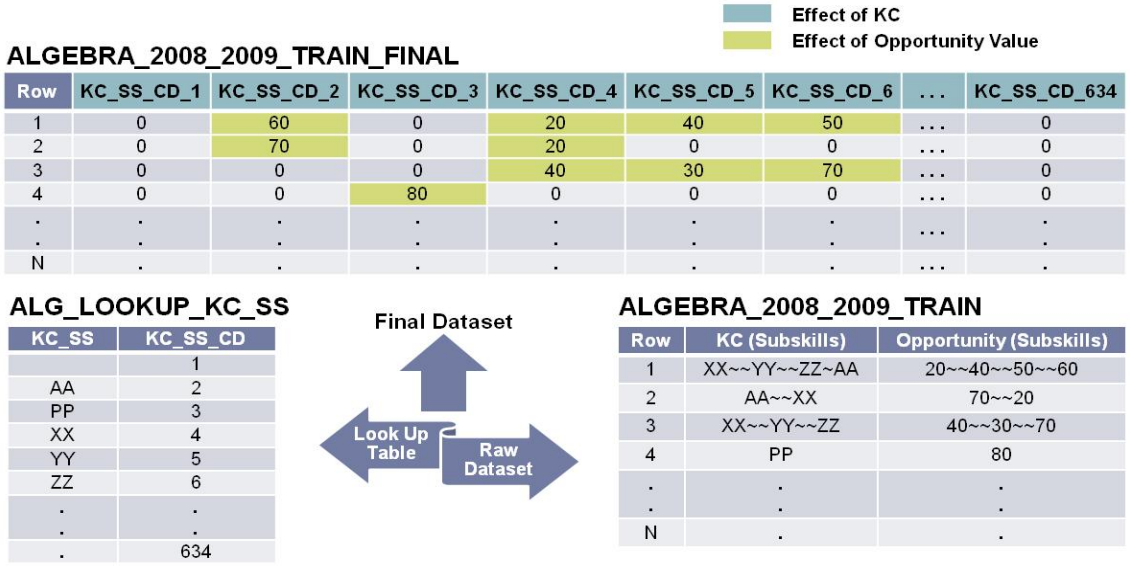


Figure 3: KC variables with opportunity values as weights

Corresponding to every knowledge component, there is an opportunity value that represents the number of times a student has encountered a step with same knowledge component.

- Knowledge component is one of the proxies for a step's difficulty level.
- Opportunity value is a proxy for student's experience that is useful to estimate his learning curve.

As last step, a final dataset is created with variables for each KC weighted by corresponding opportunity value. Refer Figure 3.

### 4.2 Step Name Text Mining

Step names have lot of unique values. Only about 8% of all step names are common to both test and train in *bridge to algebra* system dataset. For *algebra* system dataset, this number is even as low as just 3% (refer Table 1). Creation of dummies for step names, therefore, does not seem to be meaningful.

| Tutoring System | Train only | Test only | Both Train and Test | Total |
|---|---|---|---|---|
| Algebra | 675,679 | 24,735 | 19,995 | 720,409 |
| Bridge to Algebra | 115,856 | 2,047 | 10,673 | 128,576 |

Table 1: Number of unique values of step name across two tutoring systems

To make appropriate usage of given information, text mining techniques are being applied on the step names. Based on scanning of some meaningful text and mathematical operators used in step names, the outcome is a set of 28 variables : 14 as flags for occurrence and correspondingly 14 as count of occurrence. See Table 2 for details.

| S.No. | Derived Variable | Type | For Occurrence of Text |
|---|---|---|---|
| 1. | i_SN_equal_to | Flag | '=' |
| 2. | i_SN_divide | Flag | '/' |
| 3. | i_SN_multiply | Flag | '*' |
| 4. | i_SN_power | Flag | '^' |
| 5. | i_SN_bracket | Flag | '(' |
| 6. | i_SN_plus | Flag | '+' |
| 7. | i_SN_minus | Flag | '-' |
| 8. | i_SN_A_Z | Flag | an alphabet |
| 9. | i_SN_0_9 | Flag | a numeral |
| 10. | i_SN_decimal | Flag | '.' |
| 11. | i_SN_X | Flag | 'X' |
| 12. | i_SN_Y | Flag | 'Y' |
| 13. | i_SN_SQRT | Flag | 'SQRT' |
| 14. | i_SN_inequation | Flag | '<' or '>' |
| 15. | count_SN_equal_to | Count | '=' |
| 16. | count_SN_divide | Count | '/' |
| 17. | count_SN_multiply | Count | '*' |
| 18. | count_SN_power | Count | '^' |
| 19. | count_SN_bracket | Count | '(' |
| 20. | count_SN_plus | Count | '+' |
| 21. | count_SN_minus | Count | '-' |
| 22. | count_SN_A_Z | Count | an alphabet |
| 23. | count_SN_0_9 | Count | a numeral |
| 24. | count_SN_decimal | Count | '.' |
| 25. | count_SN_X | Count | 'X' |
| 26. | count_SN_Y | Count | 'Y' |
| 27. | count_SN_SQRT | Count | 'SQRT' |
| 28. | count_SN_inequation | Count | '<' or '>' |

Table 2: Text mining of step names

For *algebra* system dataset, step name length varies from 1 to 196 (refer Table 3). For *bridge to algebra* system dataset, it ranges between 1 and 357 (both inclusive).

| **Metrics** | **Algebra** | **Bridge to Algebra** |
|---|---|---|
| Total number of records (train) | 8,918,054 | 20,012,498 |
| Number of events (train) | 7,614,730 | 17,244,034 |
| Event rate (train) | 85.4 % | 86.2 % |
| Minimum step length (train) | 1 | 1 |
| Maximum step length (train) | 196 | 357 |
| Number of unique values of step length (train) | 143 | 119 |
| Minimum step length (test) | 1 | 1 |
| Maximum step length (test) | 116 | 45 |
| Number of unique values of step length (test) | 62 | 44 |

Table 3: Step name length metrics across two tutoring systems

One hypothesis is that as step name length increases, step becomes more difficult and hence the probability of first correct attempt declines. However, it also depends on factors like type of operations carried out in step and capability of student. Figure 4 reveals that step names with greater length show high fluctuations in event rate (correct first attempt rate). However, glancing at population numbers, a graph of step name length plotted against sizing depicts that majority of step names have length equal to 4 only. Hence, step name length alone may not be a very good predictor of student performance. This may be interacted with some other features.

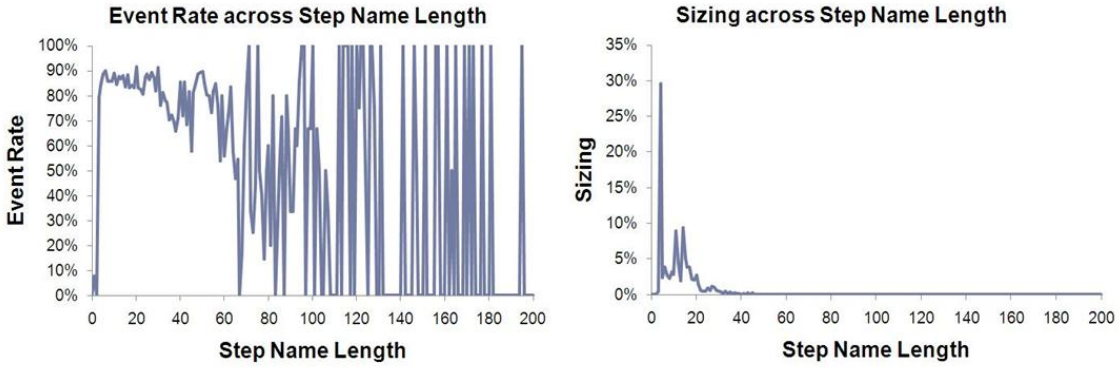


Figure 4: Event rate and sizing across length of step name

### 4.3 History-Sample Roll-Ups

This is a crucial step in feature creation methodology. Raw features have low prediction power. For most of the raw features, the linear relationship with target variable is not very

strong. As per the sampling methodology (discussed in Section 3), there exists a history dataset. Using this dataset, numerous derived variables are being created as an outcome of multi-level roll-ups. Some key variables from each roll-up are described below.

**KC level roll-ups**

- *Number of KC subskills for the step*: More number of knowledge components required for solving a step implies higher level of difficulty. This has 9 percent negative correlation with the target variable.
- *Correct first attempt response rate for the KC subskills in the step* [2]: Higher correct first attempt response rate for a knowledge component required for a step signals a low difficulty level of that step. This has a strong positive correlation of 23 percent with the target variable.
- *Average number of hints taken for the KC subskills in the step*: More hints taken for a knowledge component means less likelihood of getting a step correct in the first attempt. This has a strong negative correlation of 17 percent with the target variable.
- *Average number of corrects for the KC subskills in the step*: Higher the average number of corrects in the past for a knowledge component of step, higher the chances of future correct first attempt of steps requiring same knowledge component. This has a 7 percent positive correlation with the target variable. The relationship is not very strong as correct attempts do not directly imply correct response in the first attempt.
- *Average number of incorrects for the KC subskills in the step*: Unlike the previous variable, presence of incorrects directly implies that response at first attempt is not correct. Rolling up this historical information, the resultant variable has a strong negative correlation of 18 percent with the target variable.

**Unit level roll-ups**

- *Unit level correct first attempt response rate*: Some units are difficult for almost all students. On the contrary, some are extremely simple. For instance, in given data, while the unit of Pythagoras theorem has a distinctly very high correct first attempt response rate, it is significantly low for the unit of rational and irrational numbers. The unit level past response rate is, therefore, a good explanatory variable. It has a positive correlation of 12 percent with the target variable.
- *Unit level average of correct step duration*: Greater the amount of time spent in solving steps of a particular unit, the less likely is the chance to get them correct in the first attempt. This has a negative correlation of 8 percent.
- *Ratio of total number of correct responses for a unit to total number of attempts for that unit*: This is accuracy metric and so has a strong positive correlation of 12 percent.

2. Though this variable looks like getting created from target variable, it uses records of history dataset only. It is, therefore, a legitimate variable for modeling exercise. The logic for other such variables is analogous.

- *Ratio of total incorrect responses for a unit to total number of attempts for that unit*: Proportion of incorrect responses at unit level has a negative correlation of 11 percent.
- *Ratio of total number of hints taken for a unit to total number of attempts for that unit*: A higher proportion of hints taken for solving steps of a problem belonging to a particular unit implies lesser probability of first correct attempt. This has a negative correlation of 10 percent with the target variable.
- *Ratio of total number of correct first attempt responses for a unit to total number of correct attempts for that unit*: This is efficiency metric having a positive correlation of 9 percent with the target variable.

**Section level roll-ups**

These are analogous to unit level roll-ups. Hypotheses remain the same. Correlation coefficients are displayed below.

- *Section level correct first attempt response rate*: 14 percent positive
- *Section level average of correct step duration*: 6 percent negative
- *Ratio of total number of correct responses for a section to total number of attempts for that section*: 14 percent positive
- *Ratio of total incorrect responses for a section to total number of attempts for that section*: 13 percent negative
- *Ratio of total number of hints taken for a section to total number of attempts for that section*: 11 percent negative
- *Ratio of total number of correct first attempt responses for a unit to total number of correct attempts for that section*: 11 percent positive

**Student level roll-ups**

- *Correct first attempt response rate of student*: High response rate of student is an indicator of high proficiency level. This has a strong positive correlation of 15 percent with the target variable.
- *Ratio of total number of correct responses by student to total number of attempts by that student*: Student's accuracy metric has a positive correlation of 15 percent with the target variable.
- *Ratio of total number of correct first attempt responses by student to total number of correct attempts by that student*: Student's efficiency metric has a positive correlation of 14 percent with the target variable.
- *Total number of units attempted by student*: This is a metric of student's versatility as well as experience. It has 6 percent positive correlation with the target variable.

- *Total number of sections attempted by student*: Similar to previous variable, this is at a more granular level. This also has a positive correlation of 6 percent with the target variable.

**Step level roll-ups**

- *Average number of hints taken per step*: Current information on number of hints can't be used for developing models as it is post event information. History data has, therefore, been used to compute average number of hints taken per step. This has a negative correlation of 10 percent with the target variable.

**Problem level roll-ups**

- *Average number of hints taken per problem*: Past information on hints is being leveraged at problem level. This has strong negative correlation of 12 percent with the target variable.

### 4.4 Rasch Model

Rasch model separates the problem difficulty from the student ability. It models dichotomous response (that is, 0=incorrect and 1=correct) to a particular problem in terms of student proficiency and problem difficulty. The underlining distribution that is used is Bernoulli with a parameter which is estimated using *linear logistic test model*.

$$P_j(\theta_i) = P(X_j = 1 \mid \theta_i, \alpha_k) = \frac{1}{1 + e^{-(\theta_i - \Sigma_{k=1}^{K} q_{jk}\alpha_k)}} \tag{1}$$

where
$\theta_i$ is the proficiency of the student i,
K is the total number of skills in the transfer model being used, and
$q_{jk}$ are the entries of the transfer model
$\alpha_k$ represents the difficulty of skill k and similar to problem difficulties, higher values of $\alpha$ indicate harder skills.

To model problem difficulty, the data needs to be represented in the form of a transfer model, also known as Q-matrix:

$$\begin{pmatrix} q_{11} & \cdots & q_{1k} \\ \vdots & \ddots & \vdots \\ q_{j1} & \cdots & q_{jk} \end{pmatrix}$$

where

$$\mathrm{q}_{jk} = \begin{cases} 1, & \text{if problem } j \text{ contains skill } k \\ 0, & \text{otherwise} \end{cases}$$

Ideally better results would be obtained if the Rasch Model were developed at the most granular level which is the 'steps'. However, with the magnified unique number of steps (695,674) it is not feasible. As a substitute for step level, two levels have been considered:

- unit level (42 unique values for *algebra* and 50 unique values for *bridge to algebra*)
- problem hierarchy level (164 unique values for *algebra* and 186 unique values for *bridge to algebra*)

The dependent variable 'correct first attempt' is based on each step attempted by student. At an overall level with an event rate of 85 percent, a modified target variable is defined. For unit level analysis, it is defined as 1 if first correct attempt $\geq$ 85 percent for each unit a student attempts, else it equals zero. The transfer matrix is created for each student. If a student has not attempted a question then he gets a missing value.

Adaptive Gaussian quadrature is being implemented to get accurate approximation to MLE. The MLE coefficients represent estimated parameters for every unit's difficulty and student's proficiency simultaneously. It has been assumed that student proficiency is normally distributed.

Using NLMIXED SAS procedure, an estimate for each student's proficiency and each unit's difficulty is calculted. The correlations of the Rasch variables with the target variable are given in Table 4.

| **Level** | **Rasch Model Estimates** | **Alg. (Modeling)** | **Alg. (Validation)** | **Bri. to Alg. (Modeling)** | **Bri. to Alg. (Validation)** |
|---|---|---|---|---|---|
| Unit | Actual | 10.52% | 9.59% | 12.40% | 11.97% |
| Unit | Significant | 11.73% | 10.93% | 12.39% | 11.96% |
| Student | Actual | 13.27% | 11.52% | 10.90% | 9.49% |
| Student | Significant | 9.22% | 8.34% | 7.04% | 6.42% |

Table 4: Correlation coefficients of Rasch variables with the target variable

Significant estimates have been defined as:

$$\begin{cases} \text{Actual estimates}, & \text{if actual estimates are significant} \\ 0, & \text{if actual estimates are insignificant} \end{cases}$$

For unit level analysis, insignificant estimates of units in *algebra* system have positive sign for almost all cases and hence are set to zero. On the other hand, for student level analysis, many insignificant estimates have negative sign (particularly for *bridge to algebra* system). Actual estimates turn out to be better in case of student proficiency and hence only these have been used.

**Rasch variable has been observed as one of the significant predictors in the logistic regression model.**

### 4.5 Split Variables and Interactions

Classification trees are being constructed to identify pockets with significantly high or significantly low correct first attempt rate. Every such pocket is then represented by a binary indicator. Below are some illustrations.

**For *algebra* dataset:**

- *Indicator 1*: It takes value 1 if count of KC (subskills) > 3, else takes value 0.
- *Indicator 2*: It takes value 1 if count of KC (subskills) $\leq$ 3 and count of KC (rules) = 0, else takes value 0.
- *Indicator 3*: It takes value 1 if count of KC (rules) > 0 and count of KC (knowledge traced skills) = 0 and count of KC (subskills) $\leq$ 1, else takes value 0.

**For *bridge to algebra* dataset:**

- *Indicator 1*: It takes value 1 if count of KC (subskills) = 0 and step name contains 'X', else takes value 0.
- *Indicator 2*: It takes value 1 if count of KC (subskills) = 0 and step name does not contain '-', else takes value 0.

Several more variables have been created on the similar lines by iterating through various classification trees (post removing important variables in every successive build).

### 4.6 Mathematical Transformations

To capture non-linear relation between the target variable and the continuous independent variables, six mathematical transformations are being applied.

- Square
- Cube
- Square root
- Cube root
- Natural Logarithm
- Inverse

## 5. Features Filtration

After adding a list of advanced features to that of raw features, there is a need to undertake feature filtration process. Techniques used in the analysis are briefly mentioned in this section.

### 5.1 Correlations

For each predictor, the absolute value of its correlation coefficient with the target variable is being computed. The features with very low coefficient value are then dropped. This technique is not for relevant feature selection but for irrelevant feature elimination. With this understanding, very conservative cut-offs are being used to eliminate only a handful of variables.

### 5.2 Variable Clustering

This has been used as the second step in feature filtration process. Using PROC VARCLUS procedure in SAS, a specified list of variables gets categorized into different clusters (based on variances and correlations within and across clusters). The top two representatives are then selected from each cluster.

As a part of simultaneously applied multiple approaches, the principal component analysis has also been carried out as dimension reduction technique. However, for the given datasets, the principal component model has not added much value.

### 5.3 Variance Inflation Factor

After getting a list of features as variable clustering output, further checks are being made to remove the problem of multicollinearity. Each potential predictor is regressed on all the remaining predictors using linear regression technique. As a result, variance inflation factor (VIF) is computed for every feature. Features with VIF greater than 2 are then dropped to avoid over-fitting.

Let the final outcome list from features filtration process be termed as *eligible list of modeling variables*.

## 6. Statistical Modeling

This section is a note on statistical modeling techniques used in analysis.

### 6.1 Logistic Regression

Logistic regression technique has been used to predict the probability of a step being solved correctly in the first attempt. Following specifications have been used:

- Regression type: Binary Logistic
- Target Variable: Correct First Attempt class: 1
- Independent Variables: Subset of eligible list of modeling variables
- Selection method: Backward
- Significance level of entry: 5 percent
- Significance level of staying: 5 percent

### 6.2 CART Models

CART, a robust decision-tree tool, is being used to generate hierarchical trees such that statistically significant as well as maximum possible separation between the event and non event class is achieved at each level of split. The event rate of a terminal node is used as probability score for all records in that node. Refer Appendix B for a snapshot of CART navigator. Figure 5 shows an illustrative CART tree.

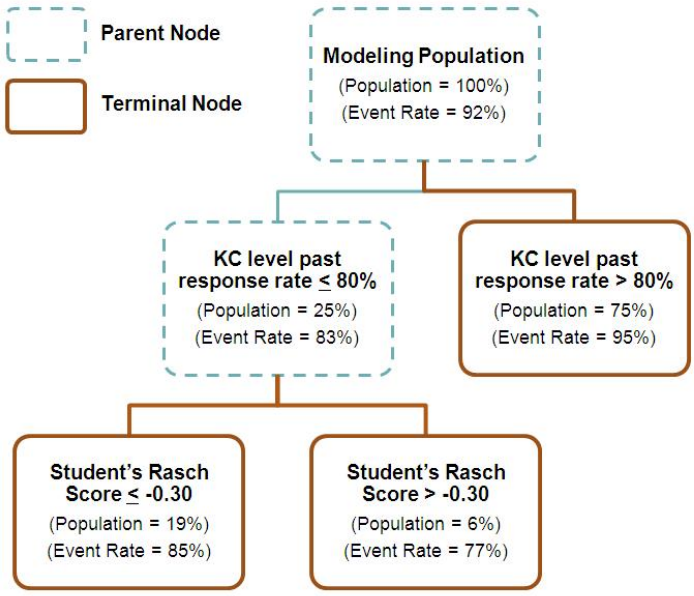


Figure 5: An illustrative CART tree

## 7. Ensemble

This section provides an overview on ensemble methodology.

### 7.1 Random Forest

With hundreds of elements in the eligible list of modeling variables, it seems unrealistic to test all the effects. To start with, a modeler may pick up a list of selected variables. However, this brings in a selection bias (for instance, one might end up choosing variables based on bi-variate profiles only). To avoid such subjective biases, random forest technique has been leveraged and implemented as follows:

- Specify modeling dataset
- Specify a list of n variables
- Draw 100 random samples of around 70
- Develop 100 logistic regression models (one model with each sample variable-list)
- For each model, compare the random input list and model outcome list of variables
- Check performance of each model in terms of RMSE value and compute the absolute difference between modeling and validation RMSE values
- Rank order models by modeling RMSE in ascending order to calculate modeling rank

- Rank order models by validation RMSE in ascending order to calculation validation rank
- Rank order models by absolute RMSE difference in ascending order to calculate stability rank
- Identify top models based on appropriate weights to all three ranks
- Take rank-weighted average of top model scores to get the final model

### 7.2 Hybrid Ensemble

Apart from averaging method, hybrid ensemble method has also been applied. Under this methodology, top models have been collated from all streams:

- Rasch model scores
- CART model score
- Random Forest model scores

Using all these scores as inputs, a hybrid logit model is developed to get the ensemble weights.

## 8. Further Explorations

There is a scope for improved models through segmentation analysis. This section illustrates the hypothesis.

### 8.1 Observation Clustering

Observation clustering seems an interesting way to identify action segments. Some select variables are first standardized with mean = 0 and standard deviation = 1. Thereafter, by applying PROC FASTCLUS procedure in SAS, 100 clusters are being created using all observations in modeling dataset. FASTCLUS statistics output dataset is being used for replicating these clusters on validation and test population. Only 7 clusters have significant population. These may be termed as segments 1-7. Based on event rates, remaining 93 clusters are being grouped into 2 major clusters: segment 8 and segment 9.

### 8.2 Segmented Models

For each of the nine segments, following steps have been followed.

- perform feature clustering to filter relevant predictors
- create 100 random samples with replacement, keeping filtered variables
- on each sample, develop a logistic regression model
- identify stable variables as those occurring in more than 90 models

- build a final logistic model using stable variables only

Adopting this methodology, nine segmented models are built. Figure 6 presents a model comparison view and helps identifying action segments.

Poor performance segments
Average performance segments
Good performance segments

| Segment | RMSE | | Absolute Difference (1)-(2) | Concordance | %Records | | | Correct First Attempt Rate | |
|---|---|---|---|---|---|---|---|---|---|
| | Modeling (1) | Validation (2) | | | Modeling | Validation | Test | Modeling | Validation |
| 1 | 0.26341 | 0.25853 | 0.00488 | 72.3 | 17.9% | 17.3% | 12.8% | 92.0% | 92.4% |
| 2 | 0.20060 | 0.20171 | 0.00111 | 71.9 | 14.2% | 14.0% | 11.0% | 95.6% | 95.6% |
| 3 | 0.27055 | 0.26501 | 0.00554 | 66.3 | 13.9% | 14.1% | 17.3% | 91.8% | 92.2% |
| 4 | 0.28764 | 0.28167 | 0.00597 | 73.0 | 13.9% | 14.5% | 18.2% | 90.0% | 90.5% |
| 5 | 0.35222 | 0.34059 | 0.01163 | 67.8 | 10.9% | 11.0% | 11.3% | 84.0% | 85.4% |
| 6 | 0.28682 | 0.27842 | 0.00840 | 72.8 | 6.6% | 6.6% | 5.1% | 88.6% | 89.2% |
| 7 | 0.31350 | 0.30782 | 0.00568 | 70.2 | 5.5% | 5.4% | 5.8% | 87.7% | 88.2% |
| 8 | 0.37967 | 0.37515 | 0.00452 | **83.0** | 7.8% | 7.8% | 9.7% | **63.3%** | **65.8%** |
| 9 | 0.11868 | 0.11543 | 0.00325 | 73.3 | 9.3% | 9.3% | 8.8% | 98.5% | 98.6% |

Figure 6: Performance of Segmented Models

Modeling and validation RMSE values are very similar, thereby corroborating high stability of models. The two models developed for segments 2 and 9 (together comprising around 25 percent population) show exceptionally good performance. The models for segments 1, 3, 4 and 6 (50 percent population) display average performance. However, models on segments 5, 7 and 8 (remaining 25 percent population) exhibit very poor performance in terms of root mean square error (RMSE).

**As a next step, if enough focus is put on these three poor performance segments, there is a great scope for improvement in overall population predictions.**

Segment 8 has very low event rate as compared to rest of the population. In entire population, there are 10 percent non-events. 26-27 percent of these are present in Segment 8 only. Another interesting point to note is that the segment 8 model is the worst performer in terms of RMSE but the best one in terms of concordance. Existing model, therefore, is good at classification. An appropriate treatment of model scores may significantly reduce RMSE value.

## 9. Software Used

SAS (Registered Trademark of the SAS Institute, Inc. of Cary, North Carolina)
CART (Salford Systems)

## Acknowledgments

We thank **Dr. Krishna Mehta** for leading the team and overall supervision. We would like to acknowledge the outstanding contribution of **Nekhil Agrawal** for his valuable inputs in advanced features creation (history variables in particular). We would also like to thank **Enakshy Dutta** for innovative analysis and successful implementation of Rasch model technique. Voluntary contributions of **Harshad Ranadive**, **Achal Gupta** and **Reena Aggarwal** are also highly acknowledged. We thank all our team members (refer Appendix A for complete list) for seamlessly working on this, despite their busy schedules.

## Appendix A.

List of all team members:

1. Abhigya Chetna
2. Achal Gupta
3. Aditya Mahajan
4. Anuja Ghosh
5. Aparna Viswanathan
6. Deepak Chopra
7. Enakshy Dutta
8. Harshad Ranadive
9. Krishna Mehta
10. Nekhil Agrawal
11. Nihit Mohan
12. Nitant Kaushal
13. Nivedita Dangwal
14. Poonam Gandhi
15. Pritish Kumar
16. Rajinder Singh Negi
17. Reena Aggarwal
18. Ruchika
19. Sachin Dean
20. Sassoon Kosian
21. Shilpi Jain
22. Sonmitra Mondal
23. Sunayna Agarwal
24. Tanu Mahajan
25. Tushar Mishra
26. Varun Aggarwal
27. Varun Kapoor

## Appendix B.

CART navigator snapshot :

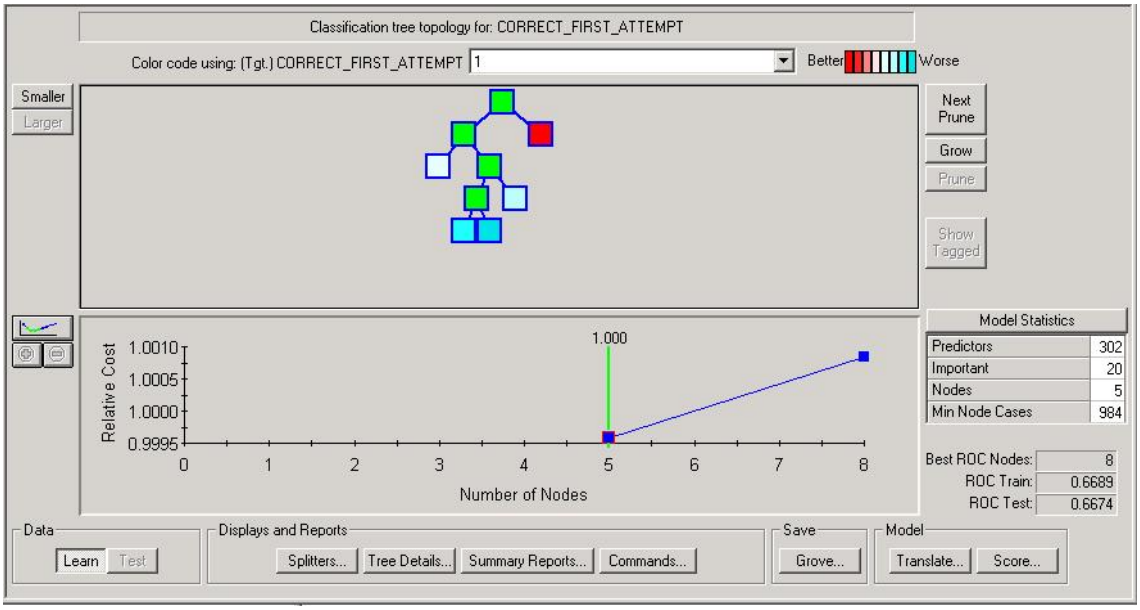


Figure 7: A snapshot of CART navigator

## References

Donna Surges Tatum. Rasch Analysis: An Introduction to Objective Measurement. *Laboratory Medicine*, 31(5): 272-274, 2000

Bryan D. Nelson. Variable Reduction for Modeling using PROC VARCLUS. *Conference Proceedings SAS Users Group International*, 261-263, 2001

G. Forman. Feature Selection: We've barely scratched the surface. *IEEE Intelligent Systems Magazine*, 20 (6): 74-76, 2005

K. R. Koedinger and V. Aleven. Exploring the assistance dilemma in experiments with Cognitive Tutors. *Educational Psychology Review*, 19 (3): 239-264, 2007

Saleema Amershi and Cristina Conati. Feature Selection: Combining Unsupervised and Supervised Classification to Build User Models for Exploratory Learning Environments. *Journal of Educational Data Mining*, 1 (1): 18-71, 2009