CS150A Course Project

Table of Contents

- Task Description
- Data Format
  - Problem
  - Step
  - Knowledge Component
  - Opportunity
  - Summary
- Evaluation
- Recommanded Coding Tools

Task Description

This project asks you to predict student performance on mathematical problems from logs of student interaction with Intelligent Tutoring Systems. This task presents interesting technical challenges, has practical importance, and is scientifically interesting.

Some student performance labels will be withheld for the test portion. You need to develop a learning model based on training data sets and use this algorithm to learn from the training portion of the challenge data sets, and then accurately predict student performance in the test sections.

Data Format

> We will offer an Ipython notebook named ***data_exploration.ipynb***, hoping this can help you understand the data format. **If you have any question about data format, please ask it on Piazza.**

> ***Suggestion***: Because of the large number of data, feature engineering may take you a long time. So, once you have finished it, use some python packages like ***Pickle*** to save it first. This will save a plenty of time for you.

Our available data takes the form of records of interactions between students and computer-aided-tutoring systems. The students solve problems in the tutor and each interaction between the student and computer is logged as a transaction. Four key terms form the building blocks of our data. These are problem, step, knowledge component, and opportunity. To more concretely define these terms, we'll use the following scenario:

Problem

A problem is a task for a student to perform that typically involves multiple steps.

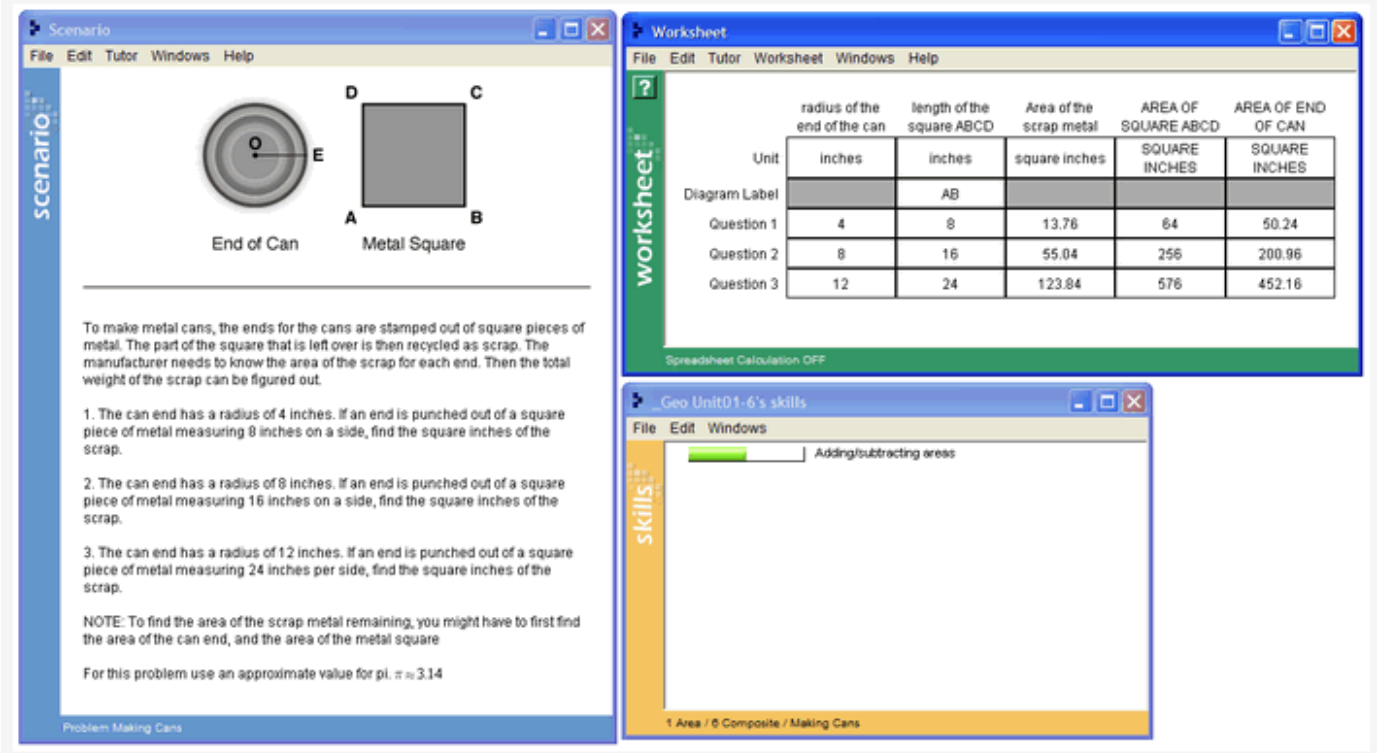

Figure 1. A problem from Carnegie Learning's Cognitive Tutor Geometry (2005 version).

In the example above, the problem asks the student to find the area of a piece of scrap metal left over after removing a circular area (the end of a can) from a metal square

Step

A step is an observable part of the solution to a problem. Because steps are observable, they are partly determined by the user interface available to the student for solving the problem. (It is not necessarily the case that the interface completely determines the steps: for example, the student might be expected to create new rows or columns of a table before filling in their entries.)

In the example problem above, the steps for the first question are:

- find the radius of the end of the can (a circle)
- find the length of the square ABCD
- find the area of the end of the can
- find the area of the square ABCD
- find the area of the left-over scrap

This whole collection of steps comprises the solution. The last step can be considered the "answer", and the others are "intermediate" steps.

Students might not (and often do not) complete a problem by performing only the correct steps—the student might request a hint from the tutor, or enter an incorrect value. We refer to the actions of a student that is working towards performing a step correctly as transactions. A transaction is an interaction between the student and the tutoring system. Each hint request, incorrect attempt, or correct attempt is a transaction, and each recorded transaction is referred to as an attempt for a step.

In Table 1, transactions have been consolidated and displayed by student and step, producing a step record table. This is the format of the data provided to you in this project. A step record is a summary of all of a given student's attempts for a given step.

Row	Student	Problem	Step	Incorrects	Hints	Error Rate	Knowledge component	Opportunity Count
1	S01	WATERING_VEGGIES	(WATERED-AREA Q1)	0	0	0	Circle-Area	1
2	S01	WATERING_VEGGIES	(TOTAL-GARDEN Q1)	2	1	1	Rectangle-Area	1
3	S01	WATERING_VEGGIES	(UNWATERED-AREA Q1)	0	0	0	Compose-Areas	1
4	S01	WATERING_VEGGIES	DONE	0	0	0	Determine-Done	1
5	S01	MAKING-CANS	(POG-RADIUS Q1)	0	0	0	Enter-Given	1
6	S01	MAKING-CANS	(SQUARE-BASE Q1)	0	0	0	Enter-Given	2
7	S01	MAKING-CANS	(SQUARE-AREA Q1)	0	0	0	Square-Area	1
8	S01	MAKING-CANS	(POG-AREA Q1)	0	0	0	Circle-Area	2
9	S01	MAKING-CANS	(SCRAP-METAL-AREA Q1)	2	0	1	Compose-Areas	2
10	S01	MAKING-CANS	(POG-RADIUS Q2)	0	0	0	Enter-Given	3
11	S01	MAKING-CANS	(SQUARE-BASE Q2)	0	0	0	Enter-Given	4
12	S01	MAKING-CANS	(SQUARE-AREA Q2)	0	0	0	Square-Area	2
13	S01	MAKING-CANS	(POG-AREA Q2)	0	0	0	Circle-Area	3
14	S01	MAKING-CANS	(SCRAP-METAL-AREA Q2)	0	0	0	Compose-Areas	3
15	S01	MAKING-CANS	(POG-RADIUS Q3)	0	0	0	Enter-Given	5
16	S01	MAKING-CANS	(SQUARE-BASE Q3)	0	0	0	Enter-Given	6
17	S01	MAKING-CANS	(SQUARE-AREA Q3)	0	0	0	Square-Area	3
18	S01	MAKING-CANS	(POG-AREA Q3)	0	0	0	Circle-Area	4
19	S01	MAKING-CANS	(SCRAP-METAL-AREA Q3)	0	0	0	Compose-Areas	4
20	S01	MAKING-CANS	DONE	0	0	0	Determine-Done	2

Knowledge Component

A knowledge component is a piece of information that can be used to accomplish tasks, perhaps along with other knowledge components. Knowledge component is a generalization of everyday terms like concept, principle, fact, or skill, and cognitive science terms like schema, production rule, misconception, or facet.

Each step in a problem requires the student to know something, a relevant concept or skill, to perform that step correctly. In given data sets, each step can be labeled with one or more hypothesized knowledge components needed—see the last column of Table 1 for example KC labels. In line 5 of Table 1, the researcher has hypothesized that the student needs to know CIRCLE-AREA to answer (POGAREA Q1). In line 6, the COMPOSE-AREAS knowledge component is hypothesized to be needed to answer (SCRAP-METAL-AREA Q1).

Every knowledge component is associated with one or more steps. One or more knowledge components can be associated with a step. This association is typically originally defined by the problem author, but researchers can provide alternative knowledge components and associations with steps; together these are known as a Knowledge Component Model.

Opportunity

An opportunity is a chance for a student to demonstrate whether he or she has learned a given knowledge component. A student's opportunity count for a given knowledge component increases by 1 each time the student encounters a step that requires this knowledge component.

An opportunity is both a test of whether a student knows a knowledge component and a chance for the student to learn it. While students may make multiple attempts at a step or request hints from a tutor (these are transactions), the whole set of attempts are considered a single opportunity. As a student works through steps in problems, he/she will have multiple opportunities to apply or learn a knowledge component.

Summary

For the data in training sets, each record will be a step that contains the following attributes:

- **Row**: the row number : for challenge data sets, the row number in each file (train, test, and submission) is no longer taken from the original data set file. Instead, rows are renumbered within each file. So instead of 1...n rows for the training file and n+1..m rows for the test/submission file, it is now 1...n for the training file and 1...n for the test/submission file.
- **Anon Student Id**: unique, anonymous identifier for a student
- **Problem Hierarchy**: the hierarchy of curriculum levels containing the problem.
- **Problem Name**: unique identifier for a problem
- **Problem View**: the total number of times the student encountered the problem so far.
- **Step Name**: each problem consists of one or more steps (e.g., "find the area of rectangle ABCD" or "divide both sides of the equation by x"). 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**: the starting time of the step. Can be null.
- **First Transaction Time**: the time of the first transaction toward the step.
- **Correct Transaction Time**: the time of the correct attempt toward the step, if there was one.
- **Step End Time**: the time of the last transaction toward the step.
- **Step Duration (sec)**: the elapsed time of the step in seconds, calculated by adding all of the durations for transactions that were attributed to the step. Can be null (if step start time is null).
- **Correct Step Duration (sec)**: the step duration if the first attempt for the step was correct.
- **Error Step Duration (sec)**: the step duration if the first attempt for the step was an error (incorrect attempt or hint request).
- **Correct First Attempt**: the tutor's evaluation of the student's first attempt on the step—1 if correct, 0 if an error.
- **Incorrects**: total number of incorrect attempts by the student on the step.
- **Hints**: total number of hints requested by the student for the step.
- **Corrects**: total correct attempts by the student for the step. (Only increases if the step is encountered more than once.)
- **KC(KC Model Name)**: the 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 ~~.
- **Opportunity(KC Model Name)**: a count that increases by one each time the student encounters a step with the listed knowledge component. Steps with multiple KCs will have multiple opportunity numbers separated by ~~.
- Additional KC models, which exist for the challenge data sets, will appear as additional pairs of columns (KC and Opportunity columns for each model).

For the test portion of the challenge data sets, values will not be provided for the following columns:

- Step Start Time
- First Transaction Time
- Correct Transaction Time
- Step End Time
- Step Duration (sec)
- Correct Step Duration (sec)
- Error Step Duration (sec)
- Correct First Attempt
- Incorrects
- Hints
- Corrects

Each data set will be split as follows:

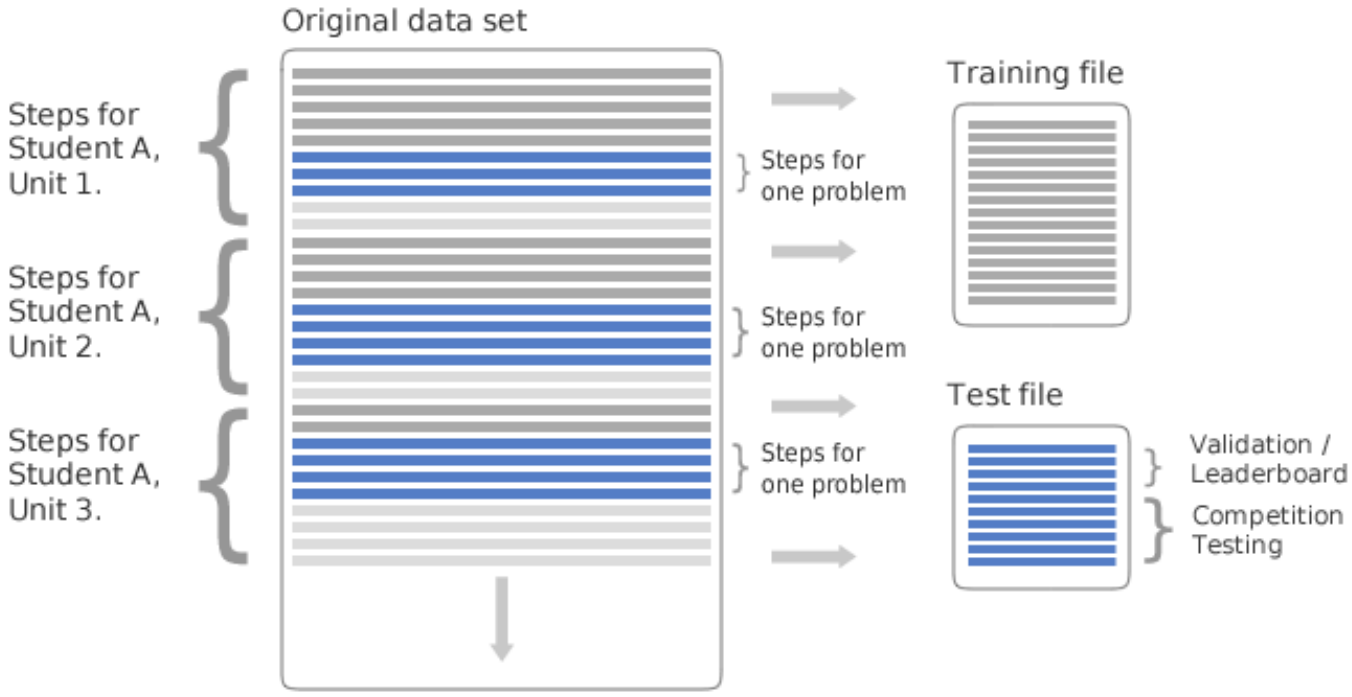

Diagram showing how data set is split into training and test files.

In the diagram above, each horizontal line represents a student-step (a record of a student working on a step.) The data set is broken down by student, unit (a classification of a portion of the math curriculum hierarchy, e.g., "Linear Inequality Graphing"), section (a portion of the curriculum that falls within a unit, e.g., "Section 1 of 3"), and problem.

Test rows are determined by a program that randomly selects one problem for each student within a unit, and places all student-step rows for that student and problem in the test file. Based on time, all preceding student-step rows for the unit will be placed in a training file, while all following student-step rows for that unit will be discarded. The goal at testing time will be to predict whether the student got the step right on the first attempt for each step in that problem. **Each prediction will take the form of a value between 0 and 1 for the column Correct First Attempt.**

Evaluation

You model will be evaluated on your performance at providing **Correct First Attempt** values for the test portion.

We will compare the predictions you provided against the undisclosed true values and report the difference as Root Mean Squared Error (RMSE). The total score for a submission will then be the average of the RMSE values.

Recommanded Coding Tools

> We recommand **Anaconda** for you, which is a platform for Python. It integrates many useful packages for data science (mentioned above) and offers a convenient package management tool -- *Conda*.

Some useful Python packages you may need:

- **Scikit-learn** (*recommanded*): a free software machine learning library.
- **Numpy** / **Scipy**: for scientific computing with Python.
- **Pandas** (*recommanded*): a library useful for reading data.
- **Matplotlib**: a Python 2D plotting library.