

Illuminate Learning. Ignite Possibilities.

Uncover new insights in early childhood education and how media can support learning outcomes. Participate in our fifth annual Data Science Bowl, presented by Booz Allen Hamilton and Kaggle.

PBS KIDS, a trusted name in early childhood education for decades, aims to gain insights into how media can help children learn important skills for success in school and life. In this challenge, you'll use anonymous gameplay data, including knowledge of videos watched and games played, from the PBS KIDS Measure Up! app, a game-based learning tool developed as a part of the CPB-PBS Ready To Learn Initiative with funding from the U.S. Department of Education. Competitors will be challenged to predict scores on in-game assessments and create an algorithm that will lead to better-designed games and improved learning outcomes. Your solutions will aid in discovering important relationships between engagement with high-quality educational media and learning processes.

Data Science Bowl is the world's largest data science competition focused on social good. Each year, this competition gives Kagglers a chance to use their passion to change the world. Over the last four years, more than 50,000+ Kagglers have submitted over 114,000+ submissions, to improve everything from lung cancer and heart disease detection to ocean health.

For more information on the Data Science Bowl, please visit DataScienceBowl.com

Where does the data for the competition come from?

The data used in this competition is anonymous, tabular data of interactions with the PBS KIDS Measure Up! app. Select data, such as a user's in-app assessment score or their path through the game, is collected by the PBS KIDS Measure Up! app, a game-based learning tool.

PBS KIDS is committed to creating a safe and secure environment that family members of all ages can enjoy. The PBS KIDS Measure Up! app does not collect any personally identifying information, such as name or location. All of the data used in the competition is anonymous. To view the full PBS KIDS privacy policy, please visit: pbskids.org/privacy.

No one will be able to download the entire data set and the participants do not have access to any personally identifiable information about individual users. The Data Science Bowl and the use of data for this year's competition has been reviewed to ensure that it meets requirements of applicable child privacy regulations by PRIVO, a leading global industry expert in children's online privacy.

What is the PBS KIDS Measure Up! app?

In the PBS KIDS Measure Up! app, children ages 3 to 5 learn early STEM concepts focused on length, width, capacity, and weight while going on an adventure through Treetop City, Magma Peak, and Crystal Caves. Joined by their favorite PBS KIDS characters, children can also collect rewards and unlock digital toys as they play. To learn more about PBS KIDS Measure Up!, please click [here](#).

PBS KIDS and the PBS KIDS Logo are registered trademarks of PBS. Used with permission. The contents of PBS KIDS Measure Up! were developed under a grant from the Department of Education. However, those contents do not necessarily represent the policy of the Department of Education, and you should not assume endorsement by the Federal Government. The app is funded by a Ready To Learn grant (PR/AWARD No. U295A150003, CFDA No. 84.295A) provided by the Department of Education to the Corporation for Public Broadcasting.

Submissions are scored based on the quadratic weighted kappa, which measures the agreement between two outcomes. This metric typically varies from 0 (random agreement) to 1 (complete agreement). In the event that there is less agreement than expected by chance, the metric may go below 0.

The outcomes in this competition are grouped into 4 groups (labeled `accuracy_group` in the data):

- 3: the assessment was solved on the first attempt
- 2: the assessment was solved on the second attempt
- 1: the assessment was solved after 3 or more attempts
- 0: the assessment was never solved

The quadratic weighted kappa is calculated as follows. First, an $N \times N$ histogram matrix O is constructed, such that $O_{i,j}$ corresponds to the number of `installation_ids` i (actual) that received a predicted value j . An N -by- N matrix of weights, w , is calculated based on the difference between actual and predicted values:

$$w_{i,j} = (i-j)^2 (N-1)^2$$

An N -by- N histogram matrix of expected outcomes, E , is calculated assuming that there is no correlation between values. This is calculated as the outer product between the actual histogram vector of outcomes and the predicted histogram vector, normalized such that E and O have the same sum.

From these three matrices, the quadratic weighted kappa is calculated as:

$$\kappa = 1 - \frac{\sum_{i,j} w_{i,j} O_{i,j}}{\sum_{i,j} w_{i,j} E_{i,j}}$$

Submission File

For each `installation_id` represented in the test set, you must predict the `accuracy_group` of the last assessment for that `installation_id`. The files must have a header and should look like the following:

```
installation_id,accuracy_group
00abae7,3
01242218,0
etc.
```

Data Description

In this dataset, you are provided with game analytics for the PBS KIDS *Measure Up!* app. In this app, children navigate a map and complete various levels, which may be activities, video clips, games, or assessments. Each assessment is designed to test a child's comprehension of a certain set of measurement-related skills. There are five assessments: Bird Measurer, Cart Balancer, Cauldron Filler, Chest Sorter, and Mushroom Sorter.

The intent of the competition is to use the gameplay data to forecast how many attempts a child will take to pass a given assessment (an incorrect answer is counted as an attempt). Each application install is represented by an `installation_id`. This will typically correspond to one child, but you should expect noise from issues such as shared devices. In the training set, you are provided the full history of gameplay data. In the test set, we have truncated the history after the start event of a single assessment, chosen randomly, for which you must predict the number of attempts. Note that the training set contains many `installation_ids` which never took

assessments, whereas every `installation_id` in the test set made an attempt on at least one assessment.

The outcomes in this competition are grouped into 4 groups (labeled `accuracy_group` in the data):

- 3: the assessment was solved on the first attempt
- 2: the assessment was solved on the second attempt
- 1: the assessment was solved after 3 or more attempts
- 0: the assessment was never solved

The file `train_labels.csv` has been provided to show how these groups would be computed on the assessments in the training set. Assessment attempts are captured in `event_code` 4100 for all assessments except for Bird Measurer, which uses `event_code` 4110. If the attempt was correct, it contains `"correct": true`.

Note that this is a [synchronous rerun code competition](#) and the private test set has approximately 8MM rows. You should be mindful of memory in your notebooks to avoid submission errors.

Files

`train.csv` & `test.csv`

These are the main data files which contain the gameplay events.

- `event_id` - Randomly generated unique identifier for the event type. Maps to `event_id` column in specs table.
- `game_session` - Randomly generated unique identifier grouping events within a single game or video play session.
- `timestamp` - Client-generated datetime
- `event_data` - Semi-structured JSON formatted string containing the events parameters. Default fields are: `event_count`, `event_code`, and `game_time`; otherwise fields are determined by the event type.
- `installation_id` - Randomly generated unique identifier grouping game sessions within a single installed application instance.
- `event_count` - Incremental counter of events within a game session (offset at 1). Extracted from `event_data`.
- `event_code` - Identifier of the event 'class'. Unique per game, but may be duplicated across games. E.g. event code '2000' always identifies the 'Start Game' event for all games. Extracted from `event_data`.
- `game_time` - Time in milliseconds since the start of the game session. Extracted from `event_data`.
- `title` - Title of the game or video.
- `type` - Media type of the game or video. Possible values are: 'Game', 'Assessment', 'Activity', 'Clip'.
- `world` - The section of the application the game or video belongs to. Helpful to identify the educational curriculum goals of the media. Possible values are: 'NONE' (at the app's start screen), 'TREETOPCITY' (Length/Height), 'MAGMAPEAK' (Capacity/Displacement), 'CRYSTALCAVES' (Weight).

`specs.csv`

This file gives the specification of the various event types.

- `event_id` - Global unique identifier for the event type. Joins to `event_id` column in events table.

- `info` - Description of the event.
- `args` - JSON formatted string of event arguments. Each argument contains:
 - `name` - Argument name.
 - `type` - Type of the argument (`string`, `int`, `number`, `object`, `array`).
 - `info` - Description of the argument.

`train_labels.csv`

This file demonstrates how to compute the ground truth for the assessments in the training set.

`sample_submission.csv`

A sample submission in the correct format.