Package: ACED (via r-universe)

July 21, 2026

Type Package

Title The ACED data set

Version 1.0.0

Description This packages the data from the ACED project.

License Artistic-2.0

Encoding UTF-8

LazyData true

Config/roxygen2/version 8.0.0

Repository https://ralmond.r-universe.dev

Date/Publication 2026-07-21 15:10:02 UTC

RemoteUrl https://github.com/ralmond/ACED

RemoteRef HEAD

RemoteSha 4b0b3c722cb0b86a63af46640f9bd67657816701

Contents

`ACED.scores` *Data from ACED field trial*

Description

ACED (Adaptive Content with Evidence-Based Diagnosis; Shute, Hansen and Almond, 2008) is a Bayes net based assessment system which featured: (a) adaptive item selection and (b) extended feedback for incorrect items. This data contains both item level and pretest/posttest data from a field trial of the ACED system.

Usage

```
data("ACED")
```

Format

ACED contains 3 primary `data.frame` objects and some supplementary data.

All of the data tables have two variables which can serve as keys. `SubjID` and `AltID`. Either can be used as a primary key in joins. Note that the first two digits of the `AltID` gives the session (i.e., class) that the student was in. Note also that students in the control group have only pretest and posttest data; hence they do not appear in `ACED.items`, `ACED.scores` or `ACED.splitHalves`.

`ACED.scores` is data frame with 230 observations on 74 variables. These are mostly high-level scores from the Bayesian network.

`Cond_code` a factor giving the experimental condition for this student, the levels are "adaptive_acc", "adaptive_full", "linear_full", and "control". Note that there are no control students in this data set.

`Sequencing` a factor describing whether the sequence of items was `Linear` or `Adaptive`

`Feedback` a factor describing whether the feedback for incorrect items was `Extended` or `AccuracyOnly`

`All_Items` a numeric vector giving the number of items in ACED

`Correct` a numeric vector giving the number of items the student got correct

`Incorr` a numeric vector giving the number of items the student got incorrect

`Remain` a numeric vector giving the number of items not reached or skipped

`ElapTime` a difftime vector giving the total time spent on ACED (in seconds) The next group of columns give "scores" for each of the nodes in the Bayesian network. Each node has four scores, and the columns are names p*nodeScoreType* where *node* is replaced by one of the short codes in `ACED.allSkills`.

p***node***`H` a numeric vector giving the probability *node* is in the high state

p***node***`M` a numeric vector giving the probability *node* is in the medium state

p***node***`L` a numeric vector giving the probability *node* is in the low state

`EAP`***node*** the expected a posteriori value of *node* assuming an equal interval scale, where `L=0`, `M=1` and `H=2`

`MAP`***node*** a factor vector giving maximum a posteriori value of *node*, i.e., `which.max(p`*node*`H, p`*node*`M, p`*node*`L)`.

`ACED.skillNames` a list with two components, `long` and `short` giving the long (spelled out in CamelCase) and short names for the skills is a character vector giving the abbreviations used for the node/skill/attributes names.

`ACED.items` is data frame with 230 observations on 73 variables. These are mostly item-level scores from the field trial. The first two columns are `SubjID` and `AltID`. The remaining columns correspond to ACED internal tasks, and are coded 1 for correct, 0 for incorrect, and `NA` for not reached.

`ACED.taskNames` is essentially the row names from `ACED.items`. The naming scheme for the tasks reflect the skills measured by the task. The response variable names all start with `t` (for task) followed by the name of one or more skills tapped by the task (if there is more than one, then the

first one is "primary".) This is followed by a numeric code, 1, 2 or 3, giving the difficulty (easy, medium or hard) and a letter (a, b or c) used to indicate alternate tasks following the same task model.

`ACED.prePost` is data frame with 290 observations on 32 variables giving the results of the pretest and posttest.

`SubjID` ID assigned by study team, "S" followed by 3 digits. Primary key.

`AltID` ID assigned by the ACED software vendor. Pattern is "sXX-YY", where *XX* is the session and *YY* is a student with the session.

`Classroom` A factor correpsonding to the student's class.

`Gender` A factor giving the student's gender (I'm not sure if this is self-report or administrative records.)

`Race` A factor giving the student's self-reported race. The codebook is lost.

`Level_Code` a factor variable describing the academic track of the student with levels `Honors`, `Academic`, `Regular`, `Part 1`, `Part 2` and `ELL`. The codes `Part 1` and `Part 2` refer to special education students in Part 1 (mainstream classroom) or Part 2 (sequestered).

`pre_scaled` scale score (after equating) on pretest

`post_scaled` scale score (after equating) on posttest

`gain_scaled` `post_scaled` - `pre_scaled`

`Form_Order` a factor variables describing whether (`AB`) Form A was the pretest and Form B was the posttest or (`BA`) vise versa.

`PreACorr` number of correct items on Form A for students who took Form A as a pretest

`PostBCorr` number of correct items on Form B for students who took Form B as a posttest

`PreBCorr` number of correct items on Form B for students who took Form B as a pretest

`PostACorr` number of correct items on Form A for students who took Form A as a posttest

`PreScore` a numeric vector with either the non-missing value from `PreACorr` and `PreBCorr`

`PostScore` a numeric vector with either the non-missing value from `PostACorr` and `PostBCorr`

`Gain` `PostScore` - `PreScore`

`preacorr_adj` `PreACorr` adjusted to put forms A and B on the same scale

`postbcorr_adj` `PostBCorr` adjusted to put forms A and B on the same scale

`prebcorr_adj` `PreBCorr` adjusted to put forms A and B on the same scale

`postacorr_adj` `PostACorr` adjusted to put forms A and B on the same scale

`Zpreacorr_adj` standardized version of `preacorr_adj`

`Zpostbcorr_adj` standardized version of `postbcorr_adj`

`Zprebcorr_adj` standardized version of `prebcorr_adj`

`Zpostacorr_adj` standardized version of `postacorr_adj`

`scale_prea` score on Form A for students who took Form A as a pretest scaled to range 0-100

`scale_preb` score on Form B for students who took Form B as a pretest scaled to range 0-100

`pre_scaled` scale score on pretest (whichever form)

`scale_posta` score on Form A for students who took Form A as a posttest scaled to range 0-100

`scale_postb` score on Form B for students who took Form B as a posttest scaled to range 0-100

`Flagged` a logical variable (codebook lost)

`Cond` a factor describing the experimental condition with levels `Adaptive/Accuracy`, `Adaptive/Extended`, `Linear/Extended` and `Control`. Note that controls are included in this data set.

`Sequencing` a factor describing whether the sequence of items was `Linear` or `Adaptive`

`Feedback` a factor describing whether the feedback for incorrect items was `Extended` or `AccuracyOnly`

`ACED.Qmatrix` is a logical matrix whose rows correspond to skills (long names) and whose columns correspond to tasks which is true if the skill is required for solving the task (according to the expert).

`ACED.QEM` is a reduced Q-matrix containing the 15 *evidence models* (unique rows in the Q-matrix). The entries are character values with "0" indicating skill not needed, "+" indicating skill is needed and "++" indicating skill is primary. The `Tasks` column lists the tasks corresponding to this evidence model (1, 2 and 3 again represent difficulty level, and the letters indicating variants). The `Anchor` column is used to identify subscales for scale identification.

`ACED.splithalves` is a list of two datasets labeled "A" and "B". Both have the same structure as `ACED.scores` (less the datas giving study condition). These were created by splitting the 62 items into 2 subforms with 31 items each. For the most part, each item was paired with an variant which differed only by the last letter. The scores are the results of Bayes net scoring with half of the items.

`ACED.pretest` and `ACED.posttest` are raw data from the external pretest and posttest given before and after the study. Each is a list with four components:

`Araw` Unscored responses for students who took that form as pre(post)test. The first row is the key.

`Ascored` The scored responses for the `Araw` students; correct is 1, incorrect is 0.

`Braw` Unscored responses for students who took that form as pre(post)test. The first row is the key.

`Bscored` The scored responses for the `Araw` students; correct is 1, incorrect is 0.

Because of the counterbalancing each student should appear in either Form A or Form B in the pretest and in the other group in the posttest. Note that the A and B forms here have no relationship with the A and B forms in `ACED.splithalves`.

Details

ACED is a Bayesian network based Assessment for Learning learning system, thus it served as both a assessment and a tutoring system. It had two novel features which could be turned on and off, elaborated feedback (turned off, it provided accuracy only feedback) and adaptive sequencing of items (turned off, it scheduled items in a fixed linear sequence).

It was originally built to cover all algebraic sequences (arithmetic, geometric and other recursive), but only the branch of the system using geometric sequences was tested. Shute, Hansen and Almond (2008) describe the field trial. Students from a local middle school (who studied arithmetic, but not geometric sequences as part of their algebra curriculum) were recruited for the study. The students were randomized into one of four groups:

`Adaptive/Accuracy` Adaptive sequencing was used, but students only received correct/incorrect feedback.

`Adaptive/Extended` Adaptive sequencing was used, but students received extended feedback for incorrect items.

`Linear/Extended` The fixed linear sequencing was used, but students received extended feedback for incorrect items.

`Control` The students did independent study and did not use ACED.

Because students in the control group were not exposed to the ACED task, neither the Bayes net level scores nor the item level scores are available for those groups, and those students are excluded from `ACED.scores` and `ACED.items`. The students are in the same order in all of the data sets, with the 60 control students tacked onto the end of the `ACED.prePost` data set.

All of the students (including the control students) were given a 25-item pretest and a 25-item posttest with items similar to the ones used in ACED. The design was counterbalanced, with half of the students receiving Form A as the pretest and Form B as the posttest and the other half the other way around, to allow the two forms to be equated using the pretest data. The details are buried in `ACED.prePost`.

Note that some irregularities were observed with the English Language Learner (`ACED.prePost$Level_code=="ELL"`) students. Their teachers were allowed to translated words for the students, but in many cases actually wound up giving instruction as part of the translation.

Source

Shute, V. J., Hansen, E. G., & Almond, R. G. (2008). You can't fatten a hog by weighing it—Or can you? Evaluating an assessment for learning system called ACED. *International Journal of Artificial Intelligence and Education*, **18**(4), 289-316.

Thanks to Val Shute for permission to use the data.

ACED development and data collection was sponsored by National Science Foundation Grant No. 0313202.

References

A more detailed description, including a Q-matrix can be found at the ECD Wiki: http://ecd.ralmond.net/ecdwiki/ACED/ACED.

See Also

`FullACED` for the full Bayes net model. `MiniACED` for a subset of the Bayes net model.

Examples

```
data(ACED)
```

FullACED *The Full ACED Model using Expert Parmeters*

Description

These are the expert models in the Peanut format. The data frames are:

FullACED.netman A manifest of networks.

FullACED.nodeman A manifest of nodes.

FullACED.omega The omega-matrix describing the competency model.

FullACED.QQ The Q-matrix describing the evidence models.

FullACED.stats A list of statistics to be reported.

Usage

```
data("FullACED")
```

Format

FullACED.netman is data frame with 36 rows (models) on the following 5 variables.

Name A character scalar, name of the network.

Title A character scalar, longer name of the network

Hub A character scalar, which compentency model this evidence model attaches to, or null for competency model.

Pathname A character scalar, the name of the file where it is stored.

Description A character scalar, a description of the network.

FullACED.nodeman is data frame with 112 rows (multiple rows per node) with the following fields:

Model A character scalar, name of the network.

NodeName A character scalar, the name of the node.

Hub A character scalar, which compentency model this evidence model attaches to, or null for competency model.

NodeTitle A character scalar, a longer name of the node.

NodeDescription A character scalar, a description of the node.

NodeLabels A character scalar, a comma separated list of labels the node has.

Continuous A logical value, is the node continuous?

Nstates An integer, number of states the node takes. Note the node will take up this many rows.

StateName Name of the state.

StateTitle,StateDescription Not used.

StateValue Effective theta value for this state.

`LowerBound`,`UpperBound` Not used.

`FullACED.omega` is a data frame with 14 rows corresponding to the competency variables and 34 columns with the following fields:

`Node` Name of the node. Note, these are the possible replacements for *XXX*.

XXX Logical vectors indicating if the variable in the column is a parent of the variable in the row.

`Link` A character vector giving the name of the link function.

`Rules` A character vector giving the name of the combination rule.

`R2` A numeric vector giving the percentage of the row variable explained by the parents.

`A.`*XXX* A numeric vector giving the weight given to the parent variable.

`Intercept` A numeric vector giving information about whether the row variable is more or less prevelant than the parents.

`PriorWeight` A numeric vector instructing how much to bias the learning algorithm towards the priors.

`FullACED.QQ` is a data frame with 35 rows corresponding to observable variables and 52 columns with the following fields:

`Model` Name of the model.

`Node` Name of the node.

`NStates` Number of states of the node.

`State` The name of the state.

`Link` The name of the link function.

`LinkScale` Unused.

XXX A logical vector 1 if the competency variable named in the column is a parent of the variable in the row.

`Rules` The name of the combination rule used.

`A.`*XXX* The weight given to the named parent.

`B.`*XXX* Not used.

`A` Not Used.

`B` The difficulty of the item.

`PriorWeight` The bias towards the prior parameters.

`FullACED.stats` is a data frame with 15 rows and 3 columns with the following fields:

`Name` A name for the statistic.

`Fun` The name of the functional to be applied.

`Node` The name of the node to which the functional will be applied.

Details

FullACED.netman and FullACED.nodeman contain the instructions for building the nets and nodes.

FullACED.omega contains instructions for building the competency model.

FullACED.QQ contains instructions for building the evidence models. Note that items whose names differ only by the last letter share the same evidence model, so there are fewer rows than items in the ACED data set.

FullACED.stats describes the statistics which are to be reported.

Source

Shute, V. J., Hansen, E. G., & Almond, R. G. (2008). You Can't Fatten A Hog by Weighing It—Or Can You? Evaluating an Assessment for Learning System Called ACED. *International Journal of Artificial Intelligence in Education*, **18**(4), 289–316. http://www.ijaied.org/iaied/ijaied/abstract/Vol_18/Shute08.html

References

An online spreadsheet with these data frames is at:

https://docs.google.com/spreadsheets/d/1oX4ndaEij_dWsap6YR_ZrQYWYUPD6urOsn4vzjBpZIQ/

See Also

MiniACED for a model subset. ACED for the response data.

Examples

```
data(FullACED)
```

MiniACED *A Subset of the ACED Model using Expert Parmeters*

Description

This is a subset of the full ACED model using only 6 competencies and 30 items.

These are the expert models in the Peanut format. The data frames are:

MiniACED.netman A manifest of networks.

MiniACED.nodeman A manifest of nodes.

MiniACED.omega The omega-matrix describing the competency model.

MiniACED.QQ The Q-matrix describing the evidence models.

MiniACED.stats A list of statistics to be reported.

Usage

```
data("MiniACED")
```

Format

`MiniACED.netman` is data frame with 36 rows (models) on the following 5 variables.

`Name` A character scalar, name of the network.

`Title` A character scalar, longer name of the network

`Hub` A character scalar, which compentency model this evidence model attaches to, or null for competency model.

`Pathname` A character scalar, the name of the file where it is stored.

`Description` A character scalar, a description of the network.

`MiniACED.nodeman` is data frame with 112 rows (multiple rows per node) with the following fields:

`Model` A character scalar, name of the network.

`NodeName` A character scalar, the name of the node.

`Hub` A character scalar, which compentency model this evidence model attaches to, or null for competency model.

`NodeTitle` A character scalar, a longer name of the node.

`NodeDescription` A character scalar, a description of the node.

`NodeLabels` A character scalar, a comma separated list of labels the node has.

`Continuous` A logical value, is the node continuous?

`Nstates` An integer, number of states the node takes. Note the node will take up this many rows.

`StateName` Name of the state.

`StateTitle`,`StateDescription` Not used.

`StateValue` Effective theta value for this state.

`LowerBound`,`UpperBound` Not used.

`MiniACED.omega` is a data frame with 14 rows corresponding to the competency variables and 34 columns with the following fields:

`Node` Name of the node. Note, these are the possible replacements for *XXX*.

XXX Logical vectors indicating if the variable in the column is a parent of the variable in the row.

`Link` A character vector giving the name of the link function.

`Rules` A character vector giving the name of the combination rule.

`R2` A numeric vector giving the percentage of the row variable explained by the parents.

`A.`*XXX* A numeric vector giving the weight given to the parent variable.

`Intercept` A numeric vector giving information about whether the row variable is more or less prevelant than the parents.

`PriorWeight` A numeric vector instructing how much to bias the learning algorithm towards the priors.

`MiniACED.QQ` is a data frame with 35 rows corresponding to observable variables and 52 columns with the following fields:

`Model` Name of the model.

`Node` Name of the node.

`NStates` Number of states of the node.

`State` The name of the state.

`Link` The name of the link function.

`LinkScale` Unused.

XXX A logical vector 1 if the competency variable named in the column is a parent of the variable in the row.

`Rules` The name of the combination rule used.

`A.`*XXX* The weight given to the named parent.

`B.`*XXX* Not used.

`A` Not Used.

`B` The difficulty of the item.

`PriorWeight` The bias towards the prior parameters.

`MiniACED.stats` is a data frame with 15 rows and 3 columns with the following fields:

`Name` A name for the statistic.

`Fun` The name of the functional to be applied.

`Node` The name of the node to which the functional will be applied.

Details

`MiniACED.netman` and `MiniACED.nodeman` contain the instructions for building the nets and nodes.

`MiniACED.omega` contains instructions for building the competency model.

`MiniACED.QQ` contains instructions for building the evidence models. Note that items whose names differ only by the last letter share the same evidence model, so there are fewer rows than items in the `ACED` data set.

`MiniACED.stats` describes the statistics which are to be reported.

Source

Shute, V. J., Hansen, E. G., & Almond, R. G. (2008). You Can't Fatten A Hog by Weighing It—Or Can You? Evaluating an Assessment for Learning System Called ACED. *International Journal of Artificial Intelligence in Education*, **18**(4), 289–316. http://www.ijaied.org/iaied/ijaied/abstract/Vol_18/Shute08.html

References

An online spreadsheet with these data frames is at:

https://docs.google.com/spreadsheets/d/1oX4ndaEij_dWsap6YR_ZrQYWYUPD6urOsn4vzjBpZIQ/

See Also

`FullACED` for the full model. `ACED` for the response data.

Examples

```
data(MiniACED)
```

Index