

# Package ‘edina’

October 13, 2022

**Title** Bayesian Estimation of an Exploratory Deterministic Input, Noisy and Gate Model

**Version** 0.1.1

**Description** Perform a Bayesian estimation of the exploratory deterministic input, noisy and gate (EDINA) cognitive diagnostic model described by Chen et al. (2018) <[doi:10.1007/s11336-017-9579-4](https://doi.org/10.1007/s11336-017-9579-4)>.

**URL** <https://github.com/tmsalab/edina>

**BugReports** <https://github.com/tmsalab/edina/issues>

**License** GPL (>= 2)

**Depends** R (>= 3.5.0)

**Imports** Rcpp, ggplot2, jjb, reshape2

**LinkingTo** Rcpp, RcppArmadillo, rgen

**Suggests** simcdm

**LazyData** true

**RoxygenNote** 7.1.0

**Encoding** UTF-8

**Language** en-US

**NeedsCompilation** yes

**Author** James Joseph Balamuta [aut, cre, cph]  
(<<https://orcid.org/0000-0003-2826-8458>>),  
Steven Andrew Culpepper [aut, cph]  
(<<https://orcid.org/0000-0003-4226-6176>>),  
Jeffrey A. Douglas [aut]

**Maintainer** James Joseph Balamuta <[balamut2@illinois.edu](mailto:balamut2@illinois.edu)>

**Repository** CRAN

**Date/Publication** 2020-03-25 13:50:15 UTC

**R topics documented:**

as_q_matrix . . . . .	2
autoplot.auto_edina . . . . .	3
auto_edina . . . . .	4
best_model . . . . .	6
BIC.edina . . . . .	7
DIC . . . . .	8
edina . . . . .	8
extract_q_matrix . . . . .	10
model_selection_graph . . . . .	11
parameter_evolution_graph . . . . .	12
PPP . . . . .	12
print.auto_edina . . . . .	13
print.edina . . . . .	14
print.q_matrix . . . . .	14
print.summary_auto_edina . . . . .	15
print.summary_edina . . . . .	16
q_graph . . . . .	16
q_matrix . . . . .	17
summary.auto_edina . . . . .	18
summary.edina . . . . .	19
<b>Index</b>	<b>20</b>

---

as_q_matrix	<i>Coerce data.frame and matrix classes to Q Matrix.</i>
-------------	--

---

**Description**

as.q\_matrix acts as an aliases.

**Usage**

```
as_q_matrix(x, ...)
```

```
## S3 method for class 'data.frame'
```

```
as_q_matrix(x, ...)
```

```
## S3 method for class 'matrix'
```

```
as_q_matrix(x, ...)
```

```
## Default S3 method:
```

```
as_q_matrix(x, ...)
```

**Arguments**

x	Either a data.frame or matrix.
...	Not used

**Value**

A q\_matrix object.'

**See Also**

[q\\_matrix\(\)](#)

**Examples**

```
# Q matrix values
x = matrix(c(1, 0, 0, 1), nrow = 2)

# Construct class
q_mat = as_q_matrix(x)
```

---

autoplot.auto\_edina    *Graph the Auto EDINA Object*

---

**Description**

Presents either the fitting of model heuristics or the evolution of parameters on a graph

**Usage**

```
## S3 method for class 'auto_edina'
autoplot(
  object,
  type = c("selection", "guessing", "slipping", "evolution"),
  ...
)
```

**Arguments**

object	An auto_edina object.
type	Kind of graph to display. Valid types: "selection" or "evolution".
...	Not used.

**Value**

A ggplot2 object.

**See Also**

[auto\\_edina\(\)](#), [best\\_model\(\)](#), [model\\_selection\\_graph\(\)](#), [parameter\\_evolution\\_graph\(\)](#)

**Examples**

```

if(requireNamespace("simcdm", quietly = TRUE)) {

  # Set a seed for reproducibility
  set.seed(1512)

  # Setup data simulation parameters
  N = 2   # Number of Examinees / Subjects
  J = 10  # Number of Items
  K = 2   # Number of Skills / Attributes

  # Note:
  # Sample size and attributes have been reduced to create a minimally
  # viable example that can be run during CRAN's automatic check.
  # Please make sure to have a larger sample size...

  # Assign slipping and guessing values for each item
  ss = gs = rep(.2, J)

  # Simulate an identifiable Q matrix
  Q = simcdm::sim_q_matrix(J, K)

  # Simulate subject attributes
  subject_alphas = simcdm::sim_subject_attributes(N, K)

  # Simulate items under the DINA model
  items_dina = simcdm::sim_dina_items(subject_alphas, Q, ss, gs)

  # Requires at least 15 seconds of execution time.
  # Three EDINA models will be fit with increasing number of attributes.
  model_set_edina = auto_edina(items_dina, k = 2:4)

  # Visualize results results
  autoplot(model_set_edina, type = "selection")

  # Equivalent to:
  model_selection_graph(model_set_edina)

  # View model parameters
  autoplot(model_set_edina, type = "guessing")

  # Or directly call with:
  parameter_evolution_graph(model_set_edina, type = "guessing")

}

```

**Description**

Automatically select an appropriate  $K$  dimension for a  $Q$  matrix under the Exploratory Deterministic Input, Noise And gate (EDINA) Model.

**Usage**

```
auto_edina(data, k = 2:4, burnin = 10000, chain_length = 20000)
```

**Arguments**

data	Binary responses to assessments in matrix form with dimensions $N \times J$ .
k	Number of Attribute Levels as a positive integer.
burnin	Number of Observations to discard on the chain.
chain_length	Length of the MCMC chain

**Value**

An auto\_edina object that contains:

- edina\_models: A list containing all estimated edina model objects.
- criteria: Information criteria calculated for each model
- k\_checked: Varying k dimensions checked.
- j: Number of Items

**See Also**

[autoplot.auto\\_edina\(\)](#), [best\\_model\(\)](#), [model\\_selection\\_graph\(\)](#), [parameter\\_evolution\\_graph\(\)](#)

**Examples**

```
if(requireNamespace("simcdm", quietly = TRUE)) {

# Set a seed for reproducibility
set.seed(1512)

# Setup data simulation parameters
N = 15 # Number of Examinees / Subjects
J = 10 # Number of Items
K = 2  # Number of Skills / Attributes

# Note:
# Sample size and attributes have been reduced to create a minimally
# viable example that can be run during CRAN's automatic check.
# Please make sure to have a larger sample size...

# Assign slipping and guessing values for each item
ss = gs = rep(.2, J)

# Simulate an identifiable Q matrix
```

```

Q = simcdm::sim_q_matrix(J, K)

# Simulate subject attributes
subject_alphas = simcdm::sim_subject_attributes(N, K)

# Simulate items under the DINA model
items_dina = simcdm::sim_dina_items(subject_alphas, Q, ss, gs)

# Requires at least 15 seconds of execution time.
# Three EDINA models will be fit with increasing number of attributes.
model_set_edina = auto_edina(items_dina, k = 2:4)

# Display results
model_set_edina

# Retrieve criterion table
table = summary(model_set_edina)

# Extract "best model"
best_model(model_set_edina)

}

```

---

best\_model

*Extract the Best Model*

---

## Description

Extracts the best model from the auto\_\*( ) search procedure.

## Usage

```
best_model(x, ...)
```

```
## S3 method for class 'auto_edina'
best_model(x, ic = c("ppp", "bic", "dic"), ...)
```

## Arguments

x	An auto_edina object
...	Not used.
ic	Information criterion name. Default "ppp".

## Value

An edina model object corresponding to the smallest value of requested information criterion.

**See Also**

[DIC.edina\(\)](#), [BIC.edina\(\)](#), [PPP.edina\(\)](#)

---

BIC.edina

*Bayesian Information Criterion (BIC)*

---

**Description**

Calculate BIC for EDINA models.

**Usage**

```
## S3 method for class 'edina'  
BIC(object, ...)
```

**Arguments**

object	An edina object
...	Not used.

**Value**

The BIC value of the given model.

**BIC Computation Procedure**

$$BIC = -2 \log p(\mathbf{y}|\hat{\theta}) + ((k + 2) * j + 2^k) \log(n)$$

**See Also**

[PPP.edina\(\)](#), [DIC.edina\(\)](#)

[PPP.edina\(\)](#), [DIC.edina\(\)](#)

---

DIC *Deviance Information Criterion (DIC)*

---

### Description

Calculate DIC for EDINA models.

### Usage

```
DIC(object, ...)
```

```
## S3 method for class 'edina'
```

```
DIC(object, ...)
```

### Arguments

object	An edina object
...	Not used.

### Value

The DIC value of the given model.

### DIC Computation Procedure

$$DIC = -2 \left( \log p(\mathbf{y}|\hat{\theta}) - 2 \left( \log p(\mathbf{y}|\hat{\theta}) - \frac{1}{N} \sum_{n=1}^N \log p(\mathbf{y}|\theta_s) \right) \right)$$

### See Also

[PPP.edina\(\)](#), [BIC.edina\(\)](#)

---

edina *EDINA Estimation Routine*

---

### Description

Performs the Exploratory Deterministic Input, Noise and Gate Model (EDINA) estimation on a given data set with a prespecified k value.

### Usage

```
edina(data, k = 3, burnin = 10000, chain_length = 20000)
```



**Arguments**

<code>data</code>	Binary responses to assessments in matrix form with dimensions $N \times J$ .
<code>k</code>	Number of Attribute Levels as a positive integer.
<code>burnin</code>	Number of Observations to discard on the chain.
<code>chain_length</code>	Length of the MCMC chain

**Value**

An edina object that contains:

- `coefficients`: Estimated coefficients of the model fit
- `loglike_summed`: Summed log-likelihood
- `loglike_pmean`: Mean of log-likelihood
- `pi_classes`: Latent classes
- `avg_q`: Estimated Averaged Q Matrix
- `est_q`: Estimated Dichotomous Q Matrix
- `or_tested`: Odds Ratio used in the model selection.
- `sample_or`: Odds Ratio for the sample.
- `n`: Number of Observations
- `j`: Number of Items
- `k`: Number of Traits
- `burnin`: Amount of iterations to discard
- `chain_length`: Amount of iterations to retain.
- `timing`: Duration of the run
- `dataset_name`: Name of the data set used in estimation.

**See Also**

[auto\\_edina\(\)](#), [summary.edina\(\)](#), [print.edina\(\)](#)

**Examples**

```
if(requireNamespace("simcdm", quietly = TRUE)) {

# Set a seed for reproducibility
set.seed(1512)

# Setup data simulation parameters
N = 1 # Number of Examinees / Subjects
J = 10 # Number of Items
K = 2 # Number of Skills / Attributes

# Note:
# Sample size and attributes have been reduced to create a minimally
# viable example that can be run during CRAN's automatic check.
```

```

# Please make sure to have a larger sample size...

# Assign slipping and guessing values for each item
ss = gs = rep(.2, J)

# Simulate an identifiable Q matrix
Q = simcdm::sim_q_matrix(J, K)

# Simulate subject attributes
subject_alphas = simcdm::sim_subject_attributes(N, K)

# Simulate items under the DINA model
items_dina = simcdm::sim_dina_items(subject_alphas, Q, ss, gs)

# Compute the edina model
edina_model = edina(items_dina, k = K)

# Display results
edina_model

# Provide a summary overview
summary(edina_model)
}

```

---

extract_q_matrix	<i>Extract Q Matrix</i>
------------------	-------------------------

---

## Description

Given a modeling object, extract the Q Matrix

## Usage

```

extract_q_matrix(x, ...)

## S3 method for class 'q_matrix'
extract_q_matrix(x, ...)

## S3 method for class 'edina'
extract_q_matrix(x, binary = TRUE, ...)

## Default S3 method:
extract_q_matrix(x, ...)

```

## Arguments

x	An edina or q_matrix object
...	Additional parameters

binary            Boolean to indicate whether the  $Q$  matrix is shown in dichotomous form or in an estimated form.

**Value**

A matrix that is either dichotomous or estimated depending on the value of the binary parameter.

**See Also**

[q\\_matrix\(\)](#), [as\\_q\\_matrix\(\)](#), [edina\(\)](#), [auto\\_edina\(\)](#)

**Examples**

```
# Q matrix values
x = matrix(c(1, 0, 0, 1), nrow = 2)

# Show Q matrix structure
Q = q_matrix(x)

# Retrieve Q matrix
extract_q_matrix(Q)
```

---

model\_selection\_graph *View Model Selection Statistics Across Models*

---

**Description**

Displays information about the value of each model information criterion for a given model across the dimensions the  $Q$  matrix is estimated.

**Usage**

```
model_selection_graph(x, ...)
```

**Arguments**

x            An auto\_edina or auto\_errum object.  
...         Not used

**Value**

A ggplot2 object

**See Also**

[autoplot.auto\\_edina\(\)](#)

---

parameter\_evolution\_graph

*View Slipping and Guessing Parameter Changes Across Models*

---

### Description

Displays the slipping and guessing parameter changes for each model across the dimensions the Q matrix is estimated.

### Usage

```
parameter_evolution_graph(x, ...)
```

### Arguments

x	An auto_edina or auto_errum object.
...	Not used

### Value

A ggplot2 object

### See Also

[autoplot.auto\\_edina\(\)](#)

---

PPP

*Posterior Predictive Probabilities (PPPs)*

---

### Description

Computes posterior predictive probabilities (PPPs) based on the odds ratios for each pair of items.

### Usage

```
PPP(object, ...)
```

```
## S3 method for class 'edina'
PPP(object, alpha = 0.05, ...)
```

### Arguments

object	An edina object
...	Not used.
alpha	Defining region to indicate the level of extremeness the data must before the model is problematic.

**Details**

PPPs that smaller than 0.05 or greater than 0.95 tend to be extreme and evidence of misfit. As a result, this is more of a heuristic metric.

**Value**

The PPP value given the specified alpha value.

**PPP Computation Procedure**

1. simulate observed responses  $\mathbf{Y}^{(r)}$  using model parameters from iteration  $r$  of the MCMC sampler
2. computing the odds ratio for each pair of items at iteration  $r$  as

$$OR^{(r)} = n_{11}^{(r)} n_{00}^{(r)} / \left( n_{10}^{(r)} n_{01}^{(r)} \right)$$

, where  $n_{11}^{(r)}$  is the frequency of ones on both variables at iteration  $r$ ,  $n_{10}^{(r)}$  is the frequency of ones on the first item and zeros on the second at iteration  $r$ , etc.; and

3. computing PPPs for each item pair as the proportion of generated  $OR^{(r)}$ 's that exceeded elements of the observed odds ratios.

---

```
print.auto_edina      Print method for auto_edina
```

---

**Description**

Custom print method for displaying the results of the Auto EDINA method.

**Usage**

```
## S3 method for class 'auto_edina'
print(x, ...)
```

**Arguments**

```
x          An auto_edina object
...        Additional values passed onto the print.data.frame method.
```

**Value**

None.

The function provides a side-effect of displaying the overview of computed results across all models estimated.

---

print.edina                      *Printing out the EDINA Object*

---

### Description

Custom print method for computing the EDINA.

### Usage

```
## S3 method for class 'edina'
print(x, binary = FALSE, ...)
```

### Arguments

x	An edina object
binary	Boolean to indicate whether the $Q$ matrix is shown in dichotomous form or in an estimated form.
...	Additional methods passed onto the print.matrix method.

### Value

None.

The function provides a side-effect of displaying the overview of the model estimated.

---

print.q\_matrix                      *Printing out a Q Matrix Object*

---

### Description

Custom print method for the Q Matrix Object.

### Usage

```
## S3 method for class 'q_matrix'
print(x, ...)
```

### Arguments

x	An q_matrix object
...	Additional methods passed onto the print.matrix method.

### Value

An invisible matrix without the q\_matrix class displayed as a part of the output displayed.

**See Also**

`q_matrix()`, `as_q_matrix()`

**Examples**

```
# Q matrix values
x = matrix(c(1, 0, 0, 1), nrow = 2)

# Show Q matrix structure
q_matrix(x)
```

---

```
print.summary_auto_edina
      Print the auto_edina model summary
```

---

**Description**

Custom method for displaying the results of the `summary(auto_edina)`.

**Usage**

```
## S3 method for class 'summary_auto_edina'
print(x, ...)
```

**Arguments**

<code>x</code>	A <code>summary_auto_edina</code> object
<code>...</code>	Additional values passed onto the <code>print.data.frame</code> method.

**Value**

None.

The function provides a side-effect of displaying the overview of computed results across all models estimated.

---

```
print.summary_edina    Printing out the Summary EDINA Object
```

---

### Description

Custom print method for displaying the EDINA model summary information.

### Usage

```
## S3 method for class 'summary_edina'
print(x, binary = FALSE, ...)
```

### Arguments

x	A summary_edina object
binary	Boolean to indicate whether the $Q$ matrix is shown in dichotomous form or in an estimated form.
...	Past onto the print.data.frame method.

### Value

None.

The function provides a side-effect of displaying the overview of the model estimated.

---

```
q_graph                Graph Q Matrix
```

---

### Description

Provides a heatmap approach to showing the estimated binary or averaged values of the Q Matrix.

### Usage

```
q_graph(x, ...)

## S3 method for class 'auto_edina'
q_graph(x, binary = TRUE, ic = c("ppp", "bic", "dic"), ...)

## S3 method for class 'edina'
q_graph(x, binary = TRUE, ...)

## S3 method for class 'matrix'
q_graph(x, ...)

## S3 method for class 'q_matrix'
q_graph(x, ...)
```



**Arguments**

x	Either an edina, auto_edina, or q_matrix object.
...	Additional parameters not used
binary	Boolean to indicate if a classified Q (dichotomous by decision rule) or an estimate Q (non-dichotomous) or should be shown. Default: TRUE.
ic	Information criterion name. Default "ppp".

**Value**

A ggplot2 object with a heatmap overview of the estimated Q matrix.

**Examples**

```
q = q_matrix(matrix(c(1, 0, 1, 1, 0, 1), ncol = 3))
q_graph(q)
```

---

q\_matrix

*Create a Q Matrix Object*

---

**Description**

Provides a way to create an object as a "q\_matrix".

**Usage**

```
q_matrix(x)
```

**Arguments**

x	Either a data.frame or matrix.
---	--------------------------------

**Value**

A q\_matrix object.

**See Also**

[as\\_q\\_matrix\(\)](#)

## Examples

```
# Q matrix values
x = matrix(c(1, 0, 0, 1), nrow = 2)

# Q matrix wrapper
q_mat = q_matrix(x)

# Data Frame encoding of Q
q_df = data.frame(
  k1 = c(1, 0),
  k2 = c(0, 1)
)

# Create a Q matrix
q_mat = q_matrix(q_df)
```

---

summary.auto\_edina      *Summarize auto\_edina model data*

---

## Description

Custom method for displaying the results of the auto\_edina.

## Usage

```
## S3 method for class 'auto_edina'
summary(object, ...)
```

## Arguments

object	An auto_edina object
...	Not used.

## Value

The original auto\_edina object with an added class of summary.auto\_edina.

---

summary.edina	<i>Summarize the EDINA Object</i>
---------------	-----------------------------------

---

**Description**

Provide a more detailed view inside of edina model object.

**Usage**

```
## S3 method for class 'edina'  
summary(object, alpha = 0.05, ...)
```

**Arguments**

object	An edina object
alpha	Defining region to indicate the level of extremeness the data must before the model is problematic.
...	Not used.

**Value**

A summary object that includes everything in the original [edina\(\)](#) object and:

- `model_fit`: Matrix of model fit summary statistics.
- `alpha`: Alpha-value used to compute [PPP\(\)](#)s.

# Index

as\_q\_matrix, 2  
as\_q\_matrix(), 11, 15, 17  
auto\_edina, 4  
auto\_edina(), 3, 9, 11  
autoplot.auto\_edina, 3  
autoplot.auto\_edina(), 5, 11, 12  
  
best\_model, 6  
best\_model(), 3, 5  
BIC.edina, 7  
BIC.edina(), 7, 8  
  
DIC, 8  
DIC.edina(), 7  
  
edina, 8  
edina(), 11, 19  
extract\_q\_matrix, 10  
  
model\_selection\_graph, 11  
model\_selection\_graph(), 3, 5  
  
parameter\_evolution\_graph, 12  
parameter\_evolution\_graph(), 3, 5  
PPP, 12  
PPP(), 19  
PPP.edina(), 7, 8  
print.auto\_edina, 13  
print.edina, 14  
print.edina(), 9  
print.q\_matrix, 14  
print.summary\_auto\_edina, 15  
print.summary\_edina, 16  
  
q\_graph, 16  
q\_matrix, 17  
q\_matrix(), 3, 11, 15  
  
summary.auto\_edina, 18  
summary.edina, 19  
summary.edina(), 9