Package 'bantha'

January 14, 2025

Title Bayesian Network Thresholding Heuristic Algorithm (BANTHA)

Version 0.1.13

Description
The BANTHA algorithm greedy search method which can be used to estimate a Bayesian network by minimizing the expected generalized structural Hamming (GSH) loss based on posterior samples from the Bayesian network distribution. See Bailey, Dahl, Andros (2024).

License MIT + file LICENSE | Apache License 2.0

Imports methods

Depends R (>= 4.2.0)

SystemRequirements Cargo (Rust's package manager), rustc (>= 1.67.1)

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.2

NeedsCompilation yes

Author David Dahl [aut, cre] (<https://orcid.org/0000-0002-8173-1547>),
Alex Crichton [ctb] (Rust crate: proc-macro2),
David B. Dahl [ctb] (Rust crates: roxido, roxido_macro),
David Tolnay [ctb] (Rust crates: proc-macro2, quote, syn, unicode-ident),
Stjepan Glavina [ctb] (Rust crate: fastrand)

Maintainer David Dahl <dahl@stat.byu.edu>

Contents

bantha-package *bantha: Bayesian Network Thresholding Heuristic Algorithm (BANTHA)*

Description

The BANTHA algorithm greedy search method which can be used to estimate a Bayesian network by minimizing the expected generalized structural Hamming (GSH) loss based on posterior samples from the Bayesian network distribution. See Bailey, Dahl, Andros (2024).

Author(s)

Maintainer: David Dahl <dahl@stat.byu.edu> (ORCID)

Other contributors:

- Alex Crichton <alex@alexcrichton.com> (Rust crate: proc-macro2) [contributor]
- David B. Dahl <dahl@stat.byu.edu> (Rust crates: roxido, roxido_macro) [contributor]
- David Tolnay <dtolnay@gmail.com> (Rust crates: proc-macro2, quote, syn, unicode-ident) [contributor]
- Stjepan Glavina <stjepang@gmail.com> (Rust crate: fastrand) [contributor]

bantha *Bayesian Network Thresholding Heuristic Algorithm (BANTHA)*

Description

This function provides a Bayesian network to summarize a network distribution using the BANTHA method (Bailey, Dahl, Andros 2024).

Usage

```
bantha(samples, a = 1, n_candidates = 0, n_cores = 0)
```

Arguments

samples	An object of class 'array' containing posterior samples from a Bayesian network distribution. Each array element encodes one network as a binary adjacency matrix, with nodes in the rows and columns. A value of 1 in cell (i, j) indicates a directed edge from node i to node j.
a	A numeric scalar for the cost parameter of generalized Hamming distance used in GSH loss. The other cost parameter, b, is equal to $2 - a$.
n_candidates	The number of starting states to use in calculating the best matrix estimate. Using all samples as starting states is recommended, this is indicated by a value of zero.
n_cores	The number of CPU cores to use, i.e., the number of simultaneous calculations at any given time. A value of zero indicates to use all cores on the system.

Value

A Bayes estimate in the form of an adjacency matrix found by minimizing the Monte Carlo approximation of the expected GSH loss using the available samples.

References

E. Bailey, D. B. Dahl, J. Andros (2024). Structure Learning of Bayesian Networks from Posterior Sample Inference. Unpublished manuscript. Available upon request from the authors.

Examples

```
data(choi25)
bantha(choi25, a = 1)
```

---

choi25 *Posterior Samples from Choi Simulation*

Description

Samples are provided from the simulated data used by Choi et al. (2020).

Usage

```
data(choi25)
```

Format

An object of class 'list' containing 1500 posterior samples from Choi et al. (2020)'s simulated posterior draws with 25 nodes. Each list element contains a binary adjacency matrix, with nodes in the rows and columns indicating the presence or absence of directed edges.

References

J. Choi, R. Chapkin, Y. Ni (2020). "Bayesian causal structural learning with zero-inflated poisson bayesian networks." Advances in Neural Information Processing Systems, 33: 5887–5897.

---

compute_expected_gsh_loss *Compute Expected Generalized Structural Hamming (GSH) Loss*

Description

This function computes the expected GSH loss (Bailey, Dahl, Andros, 2024) for a given adjacency matrix.

Usage

```
compute_expected_gsh_loss(network, gsh)
```

Arguments

network	A Bayesian network represented by an adjacency matrix.
gsh	A pointer to an initialized GSH structure containing the posterior probabilities of edges between each pair of nodes, which can be found using the initialized_expected_gsh_loss function.

Value

The expected GSH loss for the given Bayesian network based on the posterior posterior probabilities of edges.

References

Bailey, E., Dahl, D. B., Andros, J. (2024). Structure Learning of Bayesian Networks from Posterior Sample Inference. Unpublished manuscript. Available upon request from the authors.

Examples

```
data(choi25)
gsh <- initialized_expected_gsh_loss(choi25, a = 1)
compute_expected_gsh_loss(choi25[,,1], gsh)
```

---

gsh_loss_builder_new *Initialize Expected Generalized Structural Hamming (GSH) Loss*

Description

Initialize Expected Generalized Structural Hamming (GSH) Loss

Usage

```
gsh_loss_builder_new(n_items, n_samples, n_candidates, a = 1)
```

Arguments

n_items	Number of nodes in each network to process.
n_samples	Number of networks to process.
n_candidates	Number of networks to sweeten.
a	A numeric scalar for the cost parameter of generalized Hamming distance used in GSH loss. The other cost parameter, b, is equal to $2 - a$.

Value

A pointer to an initialized GSH builder structure to process samples.

References

E. Bailey, D. B. Dahl, J. Andros (2024). Structure Learning of Bayesian Networks from Posterior Sample Inference. Unpublished manuscript. Available upon request from the authors.

Examples

```
data(choi25)
gsh_builder <- gsh_loss_builder_new(ncol(choi25), nrow(choi25), 4, a = 1)
```

---

gsh_loss_builder_process

Initialize Expected Generalized Structural Hamming (GSH) Loss

---

Description

Initialize Expected Generalized Structural Hamming (GSH) Loss

Usage

```
gsh_loss_builder_process(x, gsh_loss_builder)
```

Arguments

x	A network as an adjacency matrix.
gsh_loss_builder	A pointer to a GSH builder structure.

References

E. Bailey, D. B. Dahl, J. Andros (2024). Structure Learning of Bayesian Networks from Posterior Sample Inference. Unpublished manuscript. Available upon request from the authors.

Examples

```
data(choi25)
gsh_builder <- gsh_loss_builder_new(ncol(choi25), nrow(choi25), 4, a = 1)
```

---

initialized_expected_gsh_loss

Initialize Expected Generalized Structural Hamming (GSH) Loss

---

Description

This function initializes the expected GSH loss (Bailey, Dahl, Andros, 2024) based on the given samples and value of the parameter a.

Usage

```
initialized_expected_gsh_loss(samples, a = 1)
```

Arguments

samples	An object of class 'array' containing posterior samples from a Bayesian network distribution. Each array element encodes a Bayesian network as a binary adjacency matrix, with nodes in the rows and columns. A value of 1 in cell (i, j) indicates a directed edge from node i to node j.
a	A numeric scalar for the cost parameter of generalized Hamming distance used in GSH loss. The other cost parameter, b, is equal to $2 - a$.

Value

A pointer to an initialized GSH structure containing the posterior probabilities of edges between each pair of nodes, which can be found using the 'initialized_expected_gsh_loss' function.

References

E. Bailey, D. B. Dahl, J. Andros (2024). Structure Learning of Bayesian Networks from Posterior Sample Inference. Unpublished manuscript. Available upon request from the authors.

Examples

```
data(choi25)
initialized_expected_gsh_loss(choi25, a = 1)
```

---

is_dag *Check if a Graph is a Directed Acyclic Graph (DAG)*

Description

This function checks whether the provided graph is a Directed Acyclic Graph (DAG).

Usage

```
is_dag(candidate)
```

Arguments

candidate	An adjacency matrix representing a candidate graph.

Value

A logical value indicating whether or not the graph is a DAG.

Examples

```
# Example 1: This demonstration checks if a real DAG returns true.
dag_matrix <- matrix(c(0, 1, 0,
                       0, 0, 1,
                       0, 0, 0), nrow = 3, byrow = TRUE)
is_dag(dag_matrix)

# Example 2: This demonstration checks if a graph with a cycle returns false.
```

```
cycle_matrix <- matrix(c(0, 1, 0,
                         0, 0, 1,
                         1, 0, 0), nrow = 3, byrow = TRUE)
is_dag(cycle_matrix)
```

Index