

Package: FoRDM (via r-universe)

July 2, 2026

Type Package

Title Forest Many-Objective Robust Decision Making ('FoRDM')

Version 1.0.3

Maintainer Marc Djahangard <marc.djahangard@ife.uni-freiburg.de>

Description Forest Many-Objective Robust Decision Making ('FoRDM') is a R toolkit for supporting robust forest management under deep uncertainty. It provides a forestry-focused application of Many-Objective Robust Decision Making ('MORDM') to forest simulation outputs, enabling users to evaluate robustness using regret- and 'satisficing'-based measures. 'FoRDM' identifies robust solutions, generates Pareto fronts, and offers interactive 2D, 3D, and parallel-coordinate visualizations.

License MIT + file LICENSE

URL <https://github.com/mdjahan/FoRDM>,
<https://doi.org/10.1016/j.ecoinf.2026.103903>

Encoding UTF-8

Imports dplyr (>= 1.0.0), ggplot2 (>= 3.3.0), magrittr (>= 2.0.0),
tidyr (>= 1.1.0), rlang (>= 0.4.0), tibble (>= 3.0.0), plotly
(>= 4.10.0), emoa (>= 0.5.0), wesanderson (>= 0.3.7), stats

Suggests testthat (>= 3.0.0)

RoxygenNote 7.3.3

Config/testthat/edition 3

NeedsCompilation no

Author Marc Djahangard [aut, cre], Rasoul Yousefpour [aut]

Repository <https://cran.r-universe.dev>

Date/Publication 2026-07-02 04:10:02 UTC

RemoteUrl <https://github.com/cran/FoRDM>

RemoteRef HEAD

RemoteSha bbd7c9c82e55018081803e6784b2fb06a12de37

Contents

| | |
|---|---|
| build_fordm_table | 2 |
| build_objectives_regret | 3 |
| build_objectives_satisficing | 3 |
| FoRDM | 4 |
| fordm_analysis_regret | 5 |
| fordm_analysis_satisficing | 6 |
| robustness_tradeoff_analysis | 6 |
| visualize_fordm_2d | 7 |
| visualize_fordm_3d | 8 |
| visualize_fordm_parcoord | 8 |
| visualize_fordm_parcoord_management | 9 |

| | |
|--------------|-----------|
| Index | 10 |
|--------------|-----------|

| | |
|-------------------|--------------------------|
| build_fordm_table | <i>Build FoRDM Table</i> |
|-------------------|--------------------------|

Description

Transfers the provided data table into the format for FoRDM analysis. The columns that represent management, sow (state-of-the-world, scenarios), and time have to be defined. All other columns are treated as objectives.

Usage

```
build_fordm_table(data, management, sow, time)
```

Arguments

| | |
|------------|---|
| data | A data.frame containing the input data. |
| management | The name of the management column. |
| sow | The name of the state-of-the-worlds (SOW) column. |
| time | The name of the time column. |

Value

A list with the processed data for further use in the FoRDM analysis, including the input data, mapping for identification of columns and objective columns.

 build_objectives_regret

Build Objectives Data Frame for Regret Analysis

Description

Specify for which objectives regret-based FoRDM analysis should be applied. For each objective, define its name, direction, weight, time aggregation method (mean, sum, min or max), and discount rate.

Usage

```
build_objectives_regret(
  names,
  direction = rep("maximize", length(names)),
  weights = rep(1/length(names), length(names)),
  time_aggregation = rep("mean", length(names)),
  discount_rate = rep(0, length(names))
)
```

Arguments

| | |
|------------------|---|
| names | Names of objectives as the column names in the provided data. |
| direction | Direction of objective function: 'maximize' or 'minimize'. |
| weights | Relative weights (0-1) for each objective, must sum to 1. |
| time_aggregation | Time aggregation across objectives: 'mean', 'sum', 'min' or 'max'. |
| discount_rate | Annual discount rates for each objective (e.g., 0.02 means 2% per year), applied during time aggregation. |

Value

A data frame specifying objectives, directions, weights, time aggregation methods, and discount rates for use in FoRDM analysis.

 build_objectives_satisficing

Build Objectives Data Frame for Satisficing Analysis

Description

Specify information for satisficing-based FoRDM analysis. For each objective, define its name, time aggregation method (mean, sum, min or max), discount rate, threshold and direction.

Usage

```

build_objectives_satisficing(
  names,
  time_aggregation = rep("mean", length(names)),
  discount_rate = rep(0, length(names)),
  threshold,
  direction = rep("above", length(names))
)

```

Arguments

| | |
|-------------------------------|---|
| <code>names</code> | Names of objectives as the column names in the provided data. |
| <code>time_aggregation</code> | Time aggregation across objectives: 'mean', 'sum', 'min' or 'max'. |
| <code>discount_rate</code> | Discount rates for each objective (e.g., 0.02 means 2% per time step), applied during time aggregation. |
| <code>threshold</code> | Numeric value(s) defining the satisficing level for each objective. |
| <code>direction</code> | 'above' if values should meet or exceed the threshold, 'below' if they should be lower. |

Value

A data frame specifying objectives name, time aggregation method, discount rate, threshold and direction for use in satisficing FoRDM analysis.

 FoRDM

Forest Many-Objective Robust Decision Making (FoRDM)

Description

This package provides a toolkit for supporting robust forest management under deep uncertainty. It provides a forestry-focused application of Many-Objective Robust Decision Making (MORDM) to forest simulation outputs, enabling users to evaluate robustness using regret- and satisficing-based measures. FoRDM identifies robust solutions, generates Pareto fronts, and offers interactive 2D, 3D, and parallel-coordinate visualizations.

Details

Main features:

- Create FoRDM input tables from forest simulation outputs.
- Calculate robustness measures based on Regret and Satisficing approaches.
- Visualize results using 2D, 3D, or parallel-coordinate plots.
- Explore trade-offs between robustness and performance.

Author(s)

Marc Djahangard <marc.djahangard@ife.uni-freiburg.de> (author, maintainer)
 Rasoul Yousefpour (author)

See Also

browseVignettes(package = "FoRDM")

fordm_analysis_regret *FoRDM Regret-Based many-objective Robust Decision-Making Analysis*

Description

Performs a regret-based (Type II or cVaR) many-objective robustness analysis for the provided data and objectives, providing a optimal robust management and the Pareto front.

Usage

```
fordm_analysis_regret(
  fordm_table,
  objectives,
  robustness = 0.9,
  method = "regretII"
)
```

Arguments

| | |
|-------------|---|
| fordm_table | Output from build_fordm_table(). |
| objectives | Output from build_objectives_regret(). |
| robustness | Numeric (0-1) specifying the quantile of regret used to define robustness, e.g., 0.9 evaluates management performance that is at least as good as in 90% of SOWs. |
| method | the method used to evaluate robustness <ul style="list-style-type: none"> • "regretII": Regret type II (regret to best performing alternative) approach using the robustness quantile of scenario regrets. • "CVaR": Conditional Value at Risk, using the mean of the worst (1 - robustness) fraction of weighted regrets for risk-aware selection. |

Value

A list containing the results of the FoRDM analysis:

- optimal: The management strategy identified as most robust given the regret metrics.
- pareto_front: The Pareto front of robust management strategies.

fordm_analysis_satisficing

FoRDM Satisficing-Based many-objective Robust Decision-Making Analysis

Description

Performs a satisficing-based many-objective robustness analysis for the provided data and objectives, providing an optimal robust management and the Pareto front.

Usage

```
fordm_analysis_satisficing(fordm_table, objectives, robustness = 0.9)
```

Arguments

| | |
|-------------|--|
| fordm_table | Output from build_fordm_table(). |
| objectives | Output from build_objectives_satisficing(). |
| robustness | Numeric (0-1) specifying the robustness level across SOWs, e.g., 0.9 evaluates management performance that meets objectives in at least 90% of SOWs. |

Value

A list containing the FoRDM analysis results:

- `optimal`: The management strategy that balances all objectives (Euclidean distance) while meeting the robustness threshold.
- `pareto_front`: The Pareto front of robust management strategies.

robustness_tradeoff_analysis

Robustness Trade-Off Analysis (Regret-based)

Description

Analyzes what happens when you sacrifice robustness for better performance. Shows marginal benefits and losses for each objective when switching between management strategies across different robustness levels.

Usage

```
robustness_tradeoff_analysis(
  ford_m_table,
  objectives,
  robustness_min = 0,
  robustness_max = 1,
  robustness_step = 0.05
)
```

Arguments

`fordm_table` Output from `build_FoRDM_table()`.

`objectives` Output from `build_objectives_regret()`.

`robustness_min` Minimum robustness level (default: 0.0 = 0%).

`robustness_max` Maximum robustness level (default: 1.0 = 100%).

`robustness_step` Step size for robustness levels (default: 0.05 = 5%-steps).

Value

List containing the list of optimal managements at certain robustness levels, and a plot

| | |
|---------------------------------|----------------------------------|
| <code>visualize_fordm_2d</code> | <i>Visualize 2D Pareto Front</i> |
|---------------------------------|----------------------------------|

Description

Plots a 2D plot of the Pareto front of management alternatives from `FoRDM_analysis` output.

Usage

```
visualize_fordm_2d(analysis_output, x, y, ford_m_method)
```

Arguments

`analysis_output` Output list from `FoRDM_analysis_regret()` or `FoRDM_analysis_satisficing()`.

`x` Name of the objective for the x-axis (string).

`y` Name of the objective for the y-axis (string).

`fordm_method` Either "regret" or "satisficing".

Value

A `ggplot2` object showing the 2D Pareto front for the selected objectives.

| | |
|--------------------|----------------------------------|
| visualize_fordm_3d | <i>Visualize 3D Pareto Front</i> |
|--------------------|----------------------------------|

Description

Plots a 3D plot of the Pareto front of management alternatives from FoRDM_analysis output.

Usage

```
visualize_fordm_3d(analysis_output, x, y, z, fordmethod)
```

Arguments

| | |
|-----------------|--|
| analysis_output | Output from FoRDM_analysis_regret or FoRDM_analysis_satisficing. |
| x | Name of the objective for the x-axis (string). |
| y | Name of the objective for the y-axis (string). |
| z | Name of the objective for the z-axis (string). |
| fordmethod | Either "regret" or "satisficing". |

Value

A plotly object showing the 3D Pareto front for the selected objectives.

| | |
|--------------------------|---|
| visualize_fordm_parcoord | <i>Visualize a Parallel Coordinates plot of the Pareto Front for FoRDM Analysis Results</i> |
|--------------------------|---|

Description

Creates a parallel coordinates plot showing the Pareto front from FoRDM analysis.

Usage

```
visualize_fordm_parcoord(analysis_output, fordmethod)
```

Arguments

| | |
|-----------------|--|
| analysis_output | Output from fordmethod_analysis_regret() or fordmethod_analysis_satisficing(). |
| fordmethod | Either "regret" or "satisficing". |

Value

A parallel coordinates plot object.

```
visualize_fordm_parcoord_management
```

Visualize Parallel Coordinates Plot for a single selected management across all SOWs

Description

Creates a parallel coordinates plot showing SOW performance across objectives for a selected management strategy.

Usage

```
visualize_fordm_parcoord_management(  
  ford_m_table,  
  objectives,  
  ford_m_method,  
  management  
)
```

Arguments

| | |
|---------------|--|
| ford_m_table | Output from build_ford_m_table(). |
| objectives | Output from build_objectives_regret() or build_objectives_satisficing(). |
| ford_m_method | Either "regret" or "satisficing". |
| management | Character string specifying which management to visualize. |

Value

A parallel coordinates plot object.

Index

`build_fordm_table`, [2](#)
`build_objectives_regret`, [3](#)
`build_objectives_satisficing`, [3](#)

FoRDM, [4](#)
FoRDM-package (FoRDM), [4](#)
`fordm_analysis_regret`, [5](#)
`fordm_analysis_satisficing`, [6](#)

`robustness_tradeoff_analysis`, [6](#)

`visualize_fordm_2d`, [7](#)
`visualize_fordm_3d`, [8](#)
`visualize_fordm_parcoord`, [8](#)
`visualize_fordm_parcoord_management`, [9](#)