Package 'optimLanduse'

February 24, 2021

Title Robust Land-Use Optimization

Version 0.0.4

Description Robust multi-criteria land-allocation optimization that explicitly accounts for the uncertainty of the indicators in the objective function. Solves the problem of allocating scarce land to various land-use options with regard to multiple, coequal indicators. The method aims to find the land allocation that represents the indicator composition with the best possible trade-off under uncertainty. optimLanduse includes the actual optimization procedure as described by Knoke et al. (2016) <doi:10.1038/ncomms11877> and the post-hoc calculation of the portfolio performance as presented by Gosling et al. (2020) <doi:10.1007/s10457-020-00519-0>.

License MIT + file LICENSE

Encoding UTF-8

LazyData true

RoxygenNote 7.1.1

Imports lpSolveAPI (>= 5.5.2.0-17.7), tidyr (>= 1.1.2), dplyr (>= 1.0.0)

Suggests readxl, ggplot2

URL https://gitlab.gwdg.de/forest_economics_goettingen/optimlanduse

NeedsCompilation no

Author Kai Husmann [aut, cre],
Volker von Groß [aut],
Jasper Fuchs [aut],
Kai Bödeker [aut]

Maintainer Kai Husmann <kai.husmann@uni-goettingen.de>

Repository CRAN

Date/Publication 2021-02-24 10:30:02 UTC

R topics documented:

`calcPerformance` *Attach portfolio performance and distance to target*

Description

The function calculates and attaches the portfolio performance. For a comprehensive interpretation the beta values have to be grouped by their indicator in a plot (see example). Each beta describes the relative proportion to the maximum achievable (the "target") within its indicator, given the current land use distribution and the uncertainty scenario set. The lowest beta of all indicators guarantees that under a worst-case scenario, at least this proportion will be achieved across all indicators. The solveScenario() function maximizes the guaranteed performance, or minimizes the distance to the maximum possible.

Usage

```
calcPerformance(x)
```

Arguments

`x`	An optimized optimLanduse object.

Details

For further information see the supplement of Gosling et al (2020), Formula S5 (in the supplement of the paper).

Value

An optimized optimLanduse object with attached portfolio performance.

References

Gosling, E., Reith, E., Knoke T., Gerique, A., Paul, C. (2020): Exploring farmer perceptions of agroforestry via multi-objective optimisation: a test application in Eastern Panama. *Agroforestry Systems* **94**. doi: 10.1007/s10457020005190

Examples

```
require(ggplot2)
require(readxl)

dat <- read_xlsx(exampleData("exampleGosling_2020.xlsx"),
                 col_names = FALSE)
```

```
dat <- dataPreparation(dat, uncertainty = "SE", expVAL = "score")
init <- initScenario(dat, uValue = 2,
                     optimisticRule = "expectation",
                     fixDistance = NULL)
result <- solveScenario(x = init)
performance <- calcPerformance(result)

# Visualize the distance

ggplot(performance$scenarioTable,
       aes(x = indicator,
           y = performance,
           color = indicator)) +
geom_point() +
geom_hline(yintercept =
           min(performance$scenarioTable$performance),
          linetype = "dashed", color = "red") +
ylim(0, 1)
```

`dataPreparation` *Transform data to the expected format*

Description

The input data must suit to the specific expected optimLanduse format prior to initialization and optimization. This function provides the possibility to easily transform data from the commonly used form of the exemplary data `exampleData` into to the expected format. The application of this function is not mandatory if you want to transform your data yourself or if your data is not formatted as the example data. The application example on the GitLab project page provides information about the expected structure. Incomplete rows with NA-values are deleted and an error message is displayed.

Usage

```
dataPreparation(dat, uncertainty = "SE", expVAL = "mean")
```

Arguments

`dat`	Data frame or tibble in the format of the `exampleData`.
`uncertainty`	Indicates the column name of the uncertainty in the data. Typical is "SE" for standard error or "SD" for standard deviation.
`expVAL`	Indicates the column name of the expected value.

Value

A formated coefficients table with land-use options and indicator values ready for initialization via `initScenario`.

Examples

```
require(readxl)
dat <- read_xlsx(exampleData("exampleGosling_2020.xlsx"),
                 col_names = FALSE)
dat <- dataPreparation(dat, uncertainty = "SE", expVAL = "score")
```

`exampleData` *Attach examplary data*

Description

optimLanduse comes bundled with exemplary data for land-use optimization. The files can also be found on your computer in the package folder ‘./extdata‘. These examples provide some quick applications of the package for demonstration and an example of the expected structure of the data. Consider also the GitLab project page for exemplary applications of the package.

Usage

```
exampleData(fileName = "exampleGosling_2020.xlsx")
```

Arguments

`fileName`	Name of example file. See ’details’ section for further explanation of all provided examples.

Details

exampleGosling_2020.xlsx is an excerpt from Gosling et al. 2020.

Value

The path to the example file on your computer.

References

Gosling, E., Reith, E., Knoke, T. et al. Exploring farmer perceptions of agroforestry via multi-objective optimisation: a test application in Eastern Panama. Agroforest Syst 94, 2003–2020 (2020). https://doi.org/10.1007/s10457-020-00519-0

Examples

```
require(readxl)
path <- exampleData()
read_xlsx(path, col_names = FALSE)
path <- exampleData("exampleGosling_2020.xlsx")
read_xlsx(path, col_names = FALSE)
```

`initScenario` *Initialize the robust optimization*

Description

The function initializes an *optimLanduse* S3 object on the basis of a coefficients table. Please note that the coefficients table must follow the *optimLanduse* format. The expected format is explained in the example on GitLab. Usage of `dataPreparation` is recommended to ensure that the format requirements are met.

Usage

```
initScenario(
  coefTable,
  uValue = 1,
  optimisticRule = "expectation",
  fixDistance = NULL
)
```

Arguments

`coefTable`	Coefficient table in the expected *optimLanduse* format.
`uValue`	*u* Value. The uncertainty (standard deviation or standard error) is multiplied with the u value. The value therefore enables scenario analyses with differing uncertainties in relation to indicator values. Higher u values can be interpreted as a higher risk aversion of the decision maker.
`optimisticRule`	Either *expectation* or *uncertaintyAdjustedExpectation*. The rule indicates whether the optimistic outcomes of an indicator are directly reflected by their expectations or if the indicator is calculated as expectation + uncertainty when "more is better", expectation - uncertainty respectively when "less is better". An optimization based on *expectation* considers only downside risks.
`fixDistance`	A two-column table or matrix. The table must contain the best and the worst performing landuse-type of every uncertainty scenario, which is influenced by the *uValue*. The difference between these two variables reflects the uncertainty space, in other words the distance. This table can always be found (no matter if the distance is fixed or not) in result list of the *initScenario* function. By default, the distance is not fixed *NULL*. Fixing the distance allows you to change the uncertainty level, without changing the uncertainty framework. For instance, you can then relate the achieved portfolio performance, with a low uncertainty level, to a wider and constant uncertainty framework within your analysis; so the *betas* remain comparable with each other over the course of the uncertainty analysis.

Details

The aim of separating the initialization from the optimization is to save computation time in batch analysis. The separated function calls allow the user to perform multiple optimizations from one initialized object. This could save time in scenario or sensitivity analysis.

Value

An initialized optimLanduse S3 object ready for optimization.

Examples

```
require(readxl)
dat <- read_xlsx(exampleData("exampleGosling_2020.xlsx"),
                 col_names = FALSE)
dat <- dataPreparation(dat, uncertainty = "SE", expVAL = "score")
init <- initScenario(dat,
                     uValue = 2,
                     optimisticRule = "expectation",
                     fixDistance = NULL)
```

---

solveScenario *Perform the optimization*

Description

The function solves the optimization problem, specified by the initialized *optimLanduse* object.

Usage

```
solveScenario(x, digitsPrecision = 4, lowerBound = 0, upperBound = 1)
```

Arguments

x	The initialized *optimLanduse* object. See initScenario for the initialization.
digitsPrecision	Precision of the loss value.
lowerBound	Optional lower bounds for the land-use options. Must be 0 or a vector in the dimension of the land-use options.
upperBound	Optional upper bounds for the land-use options. Must be 1 or a vector in the dimension of the land-use options.

Details

The methodological background and the formulation of the optimization framework are described in Knoke et al. (2016).

Value

A solved landUse portfolio ready for export or further data processing.

References

Knoke, T., Paul, C., Hildebrandt, P. et al. (2016): Compositional diversity of rehabilitated tropical lands supports multiple ecosystem services and buffers uncertainties. *Nat Commun* **7**, 11877. doi: 10.1038/ncomms11877

Examples

```
require(readxl)
dat <- read_xlsx(exampleData("exampleGosling_2020.xlsx"),
                 col_names = FALSE)
dat <- dataPreparation(dat, uncertainty = "SE", expVAL = "score")
init <- initScenario(dat, uValue = 2,
                     optimisticRule = "expectation",
                     fixDistance = NULL)
result <- solveScenario(x = init)
```

Index