

Package ‘evaluator’

February 26, 2017

Title Information Security Quantified Risk Assessment Toolkit

Version 0.1.0

Description An open source information security strategic risk analysis toolkit based on the OpenFAIR taxonomy <<https://www2.opengroup.org/ogsys/catalog/C13K>> and risk assessment standard <<https://www2.opengroup.org/ogsys/catalog/C13G>>. Empowers an organization to perform a quantifiable, repeatable, and data-driven review of its security program.

Depends R (>= 3.3.2)

License MIT + file LICENSE

Encoding UTF-8

LazyData true

Imports dplyr, extrafont, ggalt (>= 0.4.0), ggplot2, magrittr, mc2d, pander, purrr, psych, readr, readxl, scales, stringi, tibble, tidyr, tcltk, viridis

RoxygenNote 6.0.1

Suggests DT, flexdashboard (>= 0.4), modeest, knitr, rmarkdown, shiny, testthat

VignetteBuilder knitr

URL <http://github.com/davidski/evaluator>

BugReports <https://github.com/davidski/evaluator/issues>

NeedsCompilation no

Author David Severski [aut, cre]

Maintainer David Severski <davidski@deadheaven.com>

Repository CRAN

Date/Publication 2017-02-26 19:56:53

R topics documented:

calculate_ale	2
calculate_domain_impact	3
calculate_max_losses	4
calculate_weak_domains	4
convert_qual_to_quant	5
derive_controls	5
dollar_millions	6
domains	6
domain_summary	7
encode_scenarios	7
explore_scenarios	8
generate_event_outcomes_plot	8
generate_heatmap	9
generate_report	9
generate_scatterplot	10
get_base_fontfamily	10
import_capabilities	11
import_scenarios	11
import_spreadsheet	12
load_data	12
openfair_example	13
quantitative_scenarios	13
risk_dashboard	14
run_simulations	15
sample_lm	15
scenario_summary	16
select_events	17
simulation_results	17
split_sheet	18
summarize_all	19
summarize_domains	19
summarize_scenarios	20
theme_evaluator	20
validate_scenarios	21
Index	22

calculate_ale	<i>Run an OpenFAIR simulation</i>
---------------	-----------------------------------

Description

Run an OpenFAIR simulation

Usage

```
calculate_ale(scenario, diff_samples = NULL, diff_estimates = NULL,  
             n = 10^4, title = "Untitled", verbose = FALSE)
```

Arguments

scenario	list of tef_, tc_, and LM_ l/ml/h/conf parameters
diff_samples	Sampled difficulties for the scenario (DEPRECATED)
diff_estimates	Sampled difficulties for the scenario
n	Number of simulations to run
title	Optional name of scenario
verbose	Whether to print progress indicators

Value

Dataframe of scenario name, threat_events count, loss_events count, mean TC and DIFF exceedance, and ALE samples

calculate_domain_impact

Calculate quantified impact at a domain level.

Description

Calculate quantified impact at a domain level.

Usage

```
calculate_domain_impact(domain_summary, domains)
```

Arguments

domain_summary	Data.
domains	Data.

Value

Dataframe

calculate_max_losses *Calculate maximum losses with and without outliers.*

Description

Calculate maximum losses with and without outliers.

Usage

```
calculate_max_losses(simulation_results, scenario_outliers)
```

Arguments

simulation_results
Data.
scenario_outliers
Vector of scenario_ids which are outliers

Value

Dataframe

calculate_weak_domains
Calculate control weaknesses on a domain level.

Description

Calculate control weaknesses on a domain level.

Usage

```
calculate_weak_domains(simulation_results, domains)
```

Arguments

simulation_results
Results of running the risk simulations.
domains
Domain titles and IDs as a dataframe.

Value

Control weaknesses summarized by domain.

convert_qual_to_quant *Convert qualitative ratings to quantitative estimate ranges.*

Description

Convert qualitative ratings to quantitative estimate ranges.

Usage

```
convert_qual_to_quant(qual_label, qual_type, mappings)
```

Arguments

qual_label	Dataframe of qualitative labels (label=H/M/L, etc)
qual_type	Character string of the type (tef, tc, diff, etc)
mappings	Qualitative mappings dataframe

Value

Dataframe of estimate parameters

derive_controls *Derive control difficulty parameters for a given qualitative scenario.*

Description

Given a comma separated list of control IDs in a scenario, identify the qualitative rankings associated with each scenario, convert to their quantitative parameters, and return a dataframe of the set of parameters.

Usage

```
derive_controls(labels, capabilities, mappings, id = "None")
```

Arguments

labels	Comma delimited list of qualitative labels
capabilities	Character string of the type (tef, tc, diff, etc)
mappings	Qualitative mappings dataframe
id	Character string of the type (tef, tc, diff, etc)

Value

List-wrapped dataframe of estimate parameters

dollar_millions	<i>Format dollar amounts in terms of millions of USD.</i>
-----------------	---

Description

Format dollar amounts in terms of millions of USD.

Usage

```
dollar_millions(x)
```

Arguments

x A number.

Value

String in the format of \$xM

domains	<i>Domain mappings</i>
---------	------------------------

Description

A dataset of domains and domain IDs.

Usage

```
domains
```

Format

An object of class `tbl_df` (inherits from `tbl`, `data.frame`) with 14 rows and 2 columns.

Details

domain_id abbreviated name of the domain

domain full title of the domain

domain_summary	<i>Domain-level risk summary</i>
----------------	----------------------------------

Description

A dataset of quantified information security risk, summarized at the domain level.

Usage

```
domain_summary
```

Format

An object of class `grouped_df` (inherits from `tbl_df`, `tbl`, `data.frame`) with 140000 rows and 4 columns.

Details

ale annual loss expected, in US dollars

domain full title of the domain

domain_id abbreviated name of the domain

simulation simulation id number

encode_scenarios	<i>Encode qualitative data to quantitative parameters.</i>
------------------	--

Description

Encode qualitative data to quantitative parameters.

Usage

```
encode_scenarios(scenarios, capabilities, mappings)
```

Arguments

`scenarios` Qualitative risk scenarios dataframe

`capabilities` Qualitative program capabilities dataframe

`mappings` Number of simulations for each scenario

Value

Dataframe of encoded quantitative scenarios

generate_heatmap	<i>Display a heatmap of impact by domain.</i>
------------------	---

Description

Display a heatmap of impact by domain.

Usage

```
generate_heatmap(domain_impact)
```

Arguments

domain_impact Domain impact information from calculate_domain_impact

Value

ggplot object

generate_report	<i>Generate sample analysis report.</i>
-----------------	---

Description

Generate sample analysis report.

Usage

```
generate_report(input_directory = file.path(getwd(), "data"),  
               results_directory = file.path(getwd(), "results"),  
               output_file = "risk_report.html", ...)
```

Arguments

input_directory Location of input files
results_directory Location of simulation results
output_file Name of the output file to generate.
... Any other parameters to pass straight to rmarkdown::render

Value

Default return values of the rmarkdown::render function.

`generate_scatterplot` *Display a scatterplot for a particular scenario ID.*

Description

Display a scatterplot for a particular scenario ID.

Usage

```
generate_scatterplot(simulation_results, scenario_id)
```

Arguments

`simulation_results` Simulation results
`scenario_id` ID of the scenario to display

Value

ggplot object

`get_base_fontfamily` *Determine fonts we can/should use.*

Description

Determine fonts we can/should use.

Usage

```
get_base_fontfamily()
```

Value

String of the preferred base font

import_capabilities *Import capabilities from survey spreadsheet.*

Description

Import capabilities from survey spreadsheet.

Usage

```
import_capabilities(survey_file = system.file("survey", "survey.xlsx", package = "evaluator"), domains = NULL)
```

Arguments

survey_file	Path to survey XLSX file. Defaults to a sample file if not supplied.
domains	Dataframe of domains and domain IDs. Defaults to built-in sample domains dataset.

Value

Extracted capabilities as a dataframe.

import_scenarios *Import scenarios from survey spreadsheet.*

Description

Import scenarios from survey spreadsheet.

Usage

```
import_scenarios(survey_file = system.file("survey", "survey.xlsx", package = "evaluator"), domains = NULL)
```

Arguments

survey_file	Path to survey XLSX file. Defaults to a sample file if not supplied.
domains	Dataframe of domains and domain IDs. Defaults to built-in sample domains dataset.

Value

Extracted qualitative scenarios as a dataframe.

import_spreadsheet	<i>Import the scenario spreadsheet.</i>
--------------------	---

Description

This is a wrapper function around `import_scenarios` and `import_capabilities`, calling both functions and writing the dataframes to a location on disk.

Usage

```
import_spreadsheet(survey_file = system.file("survey", "survey.xlsx", package
  = "evaluator"), domains = NULL, output_dir = file.path(getwd(), "data"))
```

Arguments

survey_file	Path to survey XLSX file. Defaults to a sample file if not supplied.
domains	Dataframe of domains and domain IDs. Defaults to built-in sample domains dataset.
output_dir	Output file directory. Defaults to a data subdirectory in the current working direction.

Value

Dataframe of generated files (capabilities.csv and scenarios.csv)

load_data	<i>Load input and results files</i>
-----------	-------------------------------------

Description

Load input and results files

Usage

```
load_data(input_directory, results_directory)
```

Arguments

input_directory	Location of input files
results_directory	Location of simulation results

Value

List of data objects

openfair_example	<i>Launch OpenFAIR demonstration web application.</i>
------------------	---

Description

Launch OpenFAIR demonstration web application.

Usage

openfair_example()

Value

Invisible NULL

quantitative_scenarios	
------------------------	--

Information security risk scenarios

Description

A dataset of quantified information security risk scenarios, with parameters describing the distribution of each input.

Usage

quantitative_scenarios

Format

An object of class `tbl_df` (inherits from `tbl`, `data.frame`) with 56 rows and 17 columns.

Details

scenario full text description of the risk scenario

scenario_id id of the scenario, primary key

tcomm full text name of threat community

domain_id domain abbreviation

controls comma separated list of control ids that apply to this scenario

lm_l loss magnitude - low

lm_ml loss magnitude - most likely

lm_h loss magnitude - high

lm_conf loss magnitude - confidence

tc_l threat capability - low
tc_ml threat capability - most likely
tc_h threat capability - high
tc_conf threat capability - confidence
tef_l threat event frequency - low
tef_ml threat event frequency - most likely
tef_h threat event frequency - high
tef_conf threat event frequency - confidence

risk_dashboard	<i>Launch a single page summary risk dashboard.</i>
----------------	---

Description

Launch a single page summary risk dashboard.

Usage

```
risk_dashboard(input_directory = "data", results_directory = "results", ...)
```

Arguments

<code>input_directory</code>	Location of input files
<code>results_directory</code>	Location of simulation results
<code>...</code>	Any other parameters to pass straight to <code>rmarkdown::render</code>

Value

Default return values of the `rmarkdown::render` function.

run_simulations	<i>Run simulations for all scenarios.</i>
-----------------	---

Description

Run simulations for all scenarios.

Usage

```
run_simulations(scenario, simulation_count = 10000L)
```

Arguments

scenario	Quantitative scenarios
simulation_count	Number of simulations for each scenario

Value

Dataframe of raw results

sample_lm	<i>Given a number of loss events and a loss distribution, calculate losses</i>
-----------	--

Description

Given a number of loss events and a loss distribution, calculate losses

Usage

```
sample_lm(n, l, ml, h, conf)
```

Arguments

n	Numer of threat events to evaluate
l	Low boundary
ml	Most likely
h	High boundary
conf	Confidence

Value

List of total loss, min/max/mean of sle

scenario_summary *Scenario-level risk summary*

Description

A dataset of quantified information security risk, summarized at the scenario level.

Usage

scenario_summary

Format

An object of class `tbl_df` (inherits from `tbl`, `data.frame`) with 56 rows and 18 columns.

Details

ale_max maximum annual loss expected
ale_median median annual loss expected
ale_var value at risk, ale
ale_var_zscore Z-score of ale VaR
domain_id domain id
loss_events_max maximum number of loss events
loss_events_mean mean number of loss events
loss_events_median median number of loss events
loss_events_min minimum number of loss events
mean_diff_exceedance mean difficulty exceedance
mean_tc_exceedance mean threat capability exceedance
mean_vuln mean vulnerability of the scenario
outlier is this scenario an outlier
scenario_id ID of the scenario
sle_max single loss expectance max
sle_mean mean single loss expectance
sle_median median single loss expectance
sle_min minimum single loss expectance

select_events	<i>Calculate the threat capability and whether the control(s) resist the attack</i>
---------------	---

Description

Calculate the threat capability and whether the control(s) resist the attack

Usage

```
select_events(n, TCestimate, DIFFsamples = NULL, DIFFestimate = NULL,
             verbose = FALSE)
```

Arguments

n	Number of threat events to evaluate
TCestimate	Threat capability estimate - df(l, m, h, conf)
DIFFsamples	Pre-sampled
DIFFestimate	Control difficulty estimate - df(l, m, h, conf)
verbose	A boolean

Value

List of number of successful attacks (i.e. loss events), mean tc exceedance (how much TC > DIFF), and mean diff exceedance (how much DIFF > TC)

simulation_results	<i>Information security risk simulation results</i>
--------------------	---

Description

A dataset containing the full results of sample Monte Carlo simulations of information security risk scenarios.

Usage

```
simulation_results
```

Format

An object of class tbl_df (inherits from tbl, data.frame) with 56000 rows and 13 columns.

Details

domain_id domain abbreviation
scenario_id id of the scenario
simulation id of the simulation
threat_events number of threat events
loss_events number of loss events occurring in the simulation
vuln percentage of threat events that result in losses
mean_tc_exceedance mean amount of TC > DIFF
mean_diff_exceedance mean amount of DIFF > TC
ale annual loss expectancy
sle_min single loss expectancy - minimum
sle_mean single loss expectancy - mean
sle_median single loss expectancy - mean
sle_max single loss expectancy - maximum

split_sheet

Split a sheet of the survey spreadsheet into capabilities or threats.

Description

Split a sheet of the survey spreadsheet into capabilities or threats.

Usage

```
split_sheet(dat, table_type = "capabilities")
```

Arguments

dat Raw sheet input from readxl.
table_type Either capabilities or threats

Value

Extracted table as a data_Frame

summarize_all	<i>Create all summary files and write to disk.</i>
---------------	--

Description

This is a wrapper function around `summarize_scenarios` and `summarize_domains`, calling both functions and writing the dataframes to a location on disk.

Usage

```
summarize_all(simulation_results, domains, results_dir = file.path(getwd(),  
  "results"))
```

Arguments

<code>simulation_results</code>	Simulation results dataframe
<code>domains</code>	Domain mappings dataframe
<code>results_dir</code>	Directory to place simulation files

Value

Simulation results summarized by domain

summarize_domains	<i>Create domain level summary of simulation results.</i>
-------------------	---

Description

Create domain level summary of simulation results.

Usage

```
summarize_domains(simulation_results, domains)
```

Arguments

<code>simulation_results</code>	Simulation results dataframe
<code>domains</code>	Domain mappings

Value

Simulation results summarized by domain

`summarize_scenarios` *Create scenario level summary of simulation results.*

Description

Create scenario level summary of simulation results.

Usage

```
summarize_scenarios(simulation_results)
```

Arguments

`simulation_results`
Simulation results dataframe

Value

Simulation results summarized by scenario

`theme_evaluator` *Default ggplot theme used by all Evaluator-supplied graphics.*

Description

Default ggplot theme used by all Evaluator-supplied graphics.

Usage

```
theme_evaluator(base_family = "BentonSansRE")
```

Arguments

`base_family` Font family

Value

ggplot theme object

validate_scenarios *Validate scenario data.*

Description

Validate scenario data.

Usage

```
validate_scenarios(scenarios, capabilities, domains, mappings)
```

Arguments

scenarios	Dataframe of scenarios
capabilities	Dataframe of capabilities
domains	Dataframe of domain mappings
mappings	Dataframe of qualitative mappings

Value

Invisible NULL.

Index

*Topic **datasets**

- domain_summary, [7](#)
- domains, [6](#)
- quantitative_scenarios, [13](#)
- scenario_summary, [16](#)
- simulation_results, [17](#)

[calculate_ale](#), [2](#)

[calculate_domain_impact](#), [3](#)

[calculate_max_losses](#), [4](#)

[calculate_weak_domains](#), [4](#)

[convert_qual_to_quant](#), [5](#)

[derive_controls](#), [5](#)

[dollar_millions](#), [6](#)

[domain_summary](#), [7](#)

[domains](#), [6](#)

[encode_scenarios](#), [7](#)

[explore_scenarios](#), [8](#)

[generate_event_outcomes_plot](#), [8](#)

[generate_heatmap](#), [9](#)

[generate_report](#), [9](#)

[generate_scatterplot](#), [10](#)

[get_base_fontfamily](#), [10](#)

[import_capabilities](#), [11](#)

[import_scenarios](#), [11](#)

[import_spreadsheet](#), [12](#)

[load_data](#), [12](#)

[openfair_example](#), [13](#)

[quantitative_scenarios](#), [13](#)

[risk_dashboard](#), [14](#)

[run_simulations](#), [15](#)

[sample_lm](#), [15](#)

[scenario_summary](#), [16](#)

[select_events](#), [17](#)

[simulation_results](#), [17](#)

[split_sheet](#), [18](#)

[summarize_all](#), [19](#)

[summarize_domains](#), [19](#)

[summarize_scenarios](#), [20](#)

[theme_evaluator](#), [20](#)

[validate_scenarios](#), [21](#)