Package: r4subrisk (via r-universe)

May 20, 2026

Title Risk Quantification Engine for Clinical Submission Readiness

Version 0.1.1

Description Quantifies submission risk using a Failure Modes and Effects Analysis (FMEA)-inspired framework (probability, impact, detectability). Builds risk registers from evidence, computes Risk Priority Numbers (RPN), classifies risk levels, and emits standardized R4SUB (R for Regulatory Submission) evidence table rows via 'r4subcore'. Supports risk mitigation tracking and trend analysis across submission milestones.

License MIT + file LICENSE

URL https://r4sub.github.io/r4subrisk/, https://github.com/R4SUB/r4subrisk

BugReports https://github.com/R4SUB/r4subrisk/issues

Depends R (>= 4.2)

Imports cli, dplyr, r4subcore, rlang, tibble

Suggests knitr, rmarkdown, testthat (>= 3.0.0)

VignetteBuilder knitr

Config/testthat/edition 3

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.3

Config/pak/sysreqs libxml2-dev

Repository https://r4sub.r-universe.dev

Date/Publication 2026-03-16 11:27:50 UTC

RemoteUrl https://github.com/r4sub/r4subrisk

RemoteRef HEAD

RemoteSha 921e527ff3d0ccac058c4175282a68f9f62388de

Contents

`apply_mitigations` *Update Risk Mitigation Status*

Description

Applies mitigation updates to a risk register. Allows updating probability, impact, detectability, status, and mitigation notes for specific risks.

Usage

```
apply_mitigations(risk_register, updates, config = risk_config_default())
```

Arguments

`risk_register`	A `risk_register` from `create_risk_register()`.
`updates`	A data.frame with at minimum column `risk_id`, plus any columns to update: `probability`, `impact`, `detectability`, `mitigation`, `status`.
`config`	A `risk_config` from `risk_config_default()`.

Value

An updated `risk_register` with recomputed RPN and risk levels.

Examples

```
risks <- data.frame(
  risk_id = c("R001", "R002"),
  description = c("Missing vars", "Bad derivation"),
  probability = c(4, 3), impact = c(5, 4), detectability = c(2, 3)
)
rr <- create_risk_register(risks)

updates <- data.frame(
```

```
  risk_id     = "R001",
  probability = 2,
  mitigation  = "Added validation check",
  status      = "mitigated"
)
rr2 <- apply_mitigations(rr, updates)
rr2
```

`classify_rpn` *Classify RPN Value into Risk Level*

Description

Classify RPN Value into Risk Level

Usage

```
classify_rpn(rpn, bands = risk_config_default()$rpn_bands)
```

Arguments

rpn	Numeric RPN score (1–125).
bands	Named list of band boundaries from `risk_config_default()`.

Value

Character risk level name.

Examples

```
classify_rpn(90)
classify_rpn(25)
classify_rpn(5)
```

`compare_risk_registers`
Compare Risk Registers (Trend Analysis)

Description

Compares two risk register snapshots and reports changes in RPN, new risks, resolved risks, and risk level transitions.

Usage

```
compare_risk_registers(before, after)
```

Arguments

before	A risk_register (earlier snapshot).
after	A risk_register (later snapshot).

Value

A list with:

- rpn_changes: tibble of risks with changed RPN
- new_risks: risk_ids present in after but not before
- resolved_risks: risk_ids present in before but not after
- level_transitions: tibble of risk level changes
- delta_mean_rpn: change in mean RPN

Examples

```
r1 <- data.frame(
  risk_id = c("R001", "R002"),
  description = c("Missing vars", "Bad derivation"),
  probability = c(4, 3), impact = c(5, 4), detectability = c(2, 3)
)
r2 <- data.frame(
  risk_id = c("R001", "R003"),
  description = c("Missing vars", "New issue"),
  probability = c(2, 3), impact = c(5, 3), detectability = c(2, 2)
)
rr1 <- create_risk_register(r1)
rr2 <- create_risk_register(r2)
compare_risk_registers(rr1, rr2)
```

---

compute_risk_scores *Compute Risk Scores from a Risk Register*

Description

Computes aggregate risk metrics from a risk register, including mean RPN, risk distribution, and overall risk score normalized to 0–1.

Usage

```
compute_risk_scores(risk_register, config = risk_config_default())
```

Arguments

risk_register	A risk_register from create_risk_register().
config	A risk_config from risk_config_default().

Value

A list of class "risk_scores" with:

- overall_risk_score: 0–1 (0 = no risk, 1 = maximum risk)
- mean_rpn: average RPN across all risks
- max_rpn: highest individual RPN
- n_risks: total risk count
- risk_distribution: tibble of counts by risk_level
- category_summary: tibble of mean RPN by category

Examples

```
risks <- data.frame(
  risk_id = c("R001", "R002"),
  description = c("Missing vars", "Bad derivation"),
  probability = c(4, 2), impact = c(5, 3), detectability = c(2, 3)
)
rr <- create_risk_register(risks)
compute_risk_scores(rr)
```

---

create_risk_register *Create a Risk Register*

Description

Builds a risk register from a user-supplied data.frame of identified risks. Validates required columns and fills defaults.

Usage

```
create_risk_register(risks, config = risk_config_default())
```

Arguments

risks	A data.frame with at minimum columns risk_id and description. Optional columns: category, probability, impact, detectability, owner, mitigation, status.
config	A risk_config from risk_config_default().

Value

A tibble of class "risk_register" with standardized columns and computed RPN values.

Examples

```
risks <- data.frame(
  risk_id     = c("R001", "R002", "R003"),
  description = c("Missing SDTM variables", "Unmapped ADaM derivations",
                  "Inconsistent define.xml"),
  category    = c("data_quality", "traceability", "documentation"),
  probability = c(4, 3, 2),
  impact      = c(5, 4, 3),
  detectability = c(2, 3, 4)
)
rr <- create_risk_register(risks)
rr
```

---

evidence_to_risks *Derive Risk Items from Evidence*

Description

Automatically generates risk items from an R4SUB evidence table. Each failing or warning indicator becomes a potential risk, with probability and impact inferred from evidence severity.

Usage

```
evidence_to_risks(
  evidence,
  config = risk_config_default(),
  include_pass = FALSE
)
```

Arguments

evidence	A validated evidence data.frame (from r4subcore).
config	A risk_config from risk_config_default().
include_pass	Logical; if TRUE, passing indicators are also included as low-risk items. Default FALSE.

Details

The mapping from evidence to risk uses:

- risk_id: derived from indicator_id + asset_id via r4subcore::hash_id()
- category: mapped from indicator_domain

- `probability`: mapped from evidence `severity` via config
- `impact`: mapped from evidence `severity` via config
- `detectability`: uses `config$default_detectability`

Multiple evidence rows for the same indicator + asset are aggregated: probability and impact use the maximum across rows.

Value

A tibble suitable for `create_risk_register()`.

Examples

```
## Not run:
risk_items <- evidence_to_risks(evidence)
rr <- create_risk_register(risk_items)

## End(Not run)
```

`print.risk_register` *Print Risk Register*

Description

Print Risk Register

Usage

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

Arguments

`x`	A `risk_register` object.
`...`	Ignored.

print.risk_scores *Print Risk Scores*

Description

Print Risk Scores

Usage

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

Arguments

x	A risk_scores object.
...	Ignored.

risk_config_default *Default Risk Configuration*

Description

Returns configuration for risk assessment including FMEA scale definitions, RPN thresholds, and risk level classification bands.

Usage

```
risk_config_default(
 rpn_bands = list(critical = c(80, 125), high = c(40, 79), medium = c(15, 39), low =
    c(1, 14)),
 evidence_severity_to_probability = c(info = 1, low = 2, medium = 3, high = 4, critical
    = 5),
 evidence_severity_to_impact = c(info = 1, low = 2, medium = 3, high = 4, critical = 5),
  default_detectability = 3
)
```

Arguments

rpn_bands	Named list of RPN band boundaries c(lower, upper). Evaluated in order; first match wins.
evidence_severity_to_probability	Named numeric vector mapping evidence severity to probability scores (1–5 scale).
evidence_severity_to_impact	Named numeric vector mapping evidence severity to impact scores (1–5 scale).
default_detectability	Default detectability score (1–5) when not explicitly provided. Lower = more detectable.

Details

The FMEA-inspired risk model uses three dimensions:

- **Probability** (1–5): likelihood of the issue occurring/persisting
- **Impact** (1–5): severity of consequence if unresolved
- **Detectability** (1–5): difficulty of detecting the issue (1 = easy, 5 = hard)

RPN = Probability x Impact x Detectability (range 1–125)

Value

A list of class `"risk_config"` with elements: `rpn_bands`, `evidence_severity_to_probability`, `evidence_severity_to_impact`, `default_detectability`.

Examples

```
cfg <- risk_config_default()
cfg$rpn_bands
```

`risk_indicator_summary` *Compute Risk Indicator Summary*

Description

Computes summary risk indicators from a risk register, similar to `r4subtrace::trace_indicator_scores()`.

Usage

```
risk_indicator_summary(risk_register)
```

Arguments

`risk_register`	A `risk_register` from `create_risk_register()`.

Value

A tibble with columns: `indicator`, `value`, `description`.

Examples

```
risks <- data.frame(
  risk_id = c("R001", "R002", "R003"),
  description = c("Missing vars", "Bad derivation", "Label mismatch"),
  probability = c(4, 2, 1), impact = c(5, 3, 2),
  detectability = c(2, 3, 1)
)
rr <- create_risk_register(risks)
risk_indicator_summary(rr)
```

`risk_register_to_evidence`

Convert Risk Register to R4SUB Evidence

Description

Emits evidence rows compatible with `r4subcore::validate_evidence()` for each risk item in the register, plus aggregate risk metric rows.

Usage

```
risk_register_to_evidence(
  risk_register,
  ctx,
  source_name = "r4subrisk",
  source_version = NULL
)
```

Arguments

`risk_register`	A `risk_register` from `create_risk_register()`.
`ctx`	An `r4sub_run_context` from `r4subcore::r4sub_run_context()`.
`source_name`	Character; the name of the evidence source.
`source_version`	Character or `NULL`; version of the source.

Value

A data.frame of evidence rows passing `r4subcore::validate_evidence()`.

Examples

```
## Not run:
library(r4subcore)
ctx <- r4sub_run_context(study_id = "TEST001", environment = "DEV")
risks <- data.frame(
  risk_id = "R001", description = "Missing vars",
```

```
  probability = 4, impact = 5, detectability = 2
)
rr <- create_risk_register(risks)
ev <- risk_register_to_evidence(rr, ctx = ctx)

## End(Not run)
```

Index