

Package ‘texanshootR’

May 30, 2026

Type Package

Title Reproducible Audit Trails for Indefensible Research

Version 0.1.0

Description Provides a structured, terminal-first interface for exploratory model search, including transformation grids, predictor-subset enumeration, interaction screening, principled-sounding sample restrictions, outcome engineering, and model-form escalation (polynomial / spline wraps, robust M-estimation, generalized linear model (GLM) family swaps, random-intercept lifts). Persistent run history, achievement tracking, and reportable output generators (manuscript, presentation, funding letter, graphical abstract, reviewer response) are included.

License MIT + file LICENSE

URL <https://gillescolling.com/texanshootR/>,
<https://github.com/gcol33/texanshootR>

BugReports <https://github.com/gcol33/texanshootR/issues>

Encoding UTF-8

RoxygenNote 7.3.3

Depends R (>= 4.1.0)

Imports cli, yaml, jsonlite, stats, utils, tools, splines, Rcpp

LinkingTo Rcpp

Suggests ggplot2, officer, rmarkdown, xml2, quarto, tinytex, withr, testthat (>= 3.0.0), knitr, pkgdown

Config/testthat/edition 3

VignetteBuilder knitr

NeedsCompilation yes

Author Gilles Colling [aut, cre] (ORCID:
<https://orcid.org/0000-0003-3070-6066>)

Maintainer Gilles Colling <gilles.colling051@gmail.com>

Repository CRAN

Date/Publication 2026-05-30 08:30:02 UTC

Contents

abstract	2
achievements	3
career	3
funding	4
graphical_abstract	4
manuscript	5
presentation	6
progress	6
reset	7
reviewer_response	8
run_log	8
shoot	9
texanshootR-options	10
validate_messages	11
wardrobe	11

Index **13**

abstract	<i>Generate a one-paragraph deadpan abstract</i>
----------	--

Description

The entry-tier output. Always the first stage of the publication chain (see [progress\(\)](#)). Writes a plain-text `.txt` file describing the highlighted specification in the unbothered register typical of an applied-stats abstract.

Usage

```
abstract(run, output_dir = NULL, file = NULL, force = FALSE)
```

Arguments

run	A <code>tx_run</code> object returned by shoot() . Must be the currently active publication chain — i.e., the run that opened the chain when it cleared $p \leq 0.05$.
output_dir	Optional output directory.
file	Optional filename stem (without extension).
force	Overwrite an existing file.

Details

Chain stage: **abstract** (length 1, always unlocked).

Value

Character path to the written file (invisible).

achievements	<i>Inspect the achievement registry</i>
--------------	---

Description

Returns a data.frame of all achievements with unlock status. Hidden achievements appear with name = "???" and hint = NA until unlocked.

Usage

```
achievements()
```

Value

A data.frame.

career	<i>Inspect career state</i>
--------	-----------------------------

Description

Returns a tx_career object describing the persistent career, or a fresh default if no save exists. The object's print method shows a compact dashboard card; summary adds aggregate counts. Career tier is derived from the publication-chain length you've unlocked (see [progress\(\)](#)); XP comes from completed chain stages.

Usage

```
career()
```

Value

A tx_career object.

funding	<i>Generate a funding letter + budget DOCX</i>
---------	--

Description

One-page Letter of Intent with a costed budget table.

Usage

```
funding(run, output_dir = NULL, file = NULL, force = FALSE)
```

Arguments

run	A tx_run object returned by shoot() .
output_dir	Optional output directory.
file	Optional filename stem (without extension).
force	Overwrite an existing file.

Details

Chain stage: **funding** (length 6). See [progress\(\)](#).

Value

A length-one character vector giving the path to the written DOCX file, returned invisibly. Called for its side effect of generating the funding letter in output_dir and advancing the publication chain.

graphical_abstract	<i>Generate a graphical abstract PNG</i>
--------------------	--

Description

A single-figure summary of the highlighted specification, composed as a multi-panel layout with conceptual arrows. Renders via ggplot2 to PNG.

Usage

```
graphical_abstract(run, output_dir = NULL, file = NULL, force = FALSE)
```

Arguments

run	A tx_run object returned by shoot() .
output_dir	Optional output directory.
file	Optional filename stem (without extension).
force	Overwrite an existing file.

Details

Chain stage: **graphical_abstract** (length 5). See [progress\(\)](#).

Value

A length-one character vector giving the path to the written PNG file, returned invisibly. Called for its side effect of generating the graphical abstract in `output_dir` and advancing the publication chain.

manuscript

Generate a manuscript from a run

Description

Renders an academic-style write-up of the highlighted specification. If `quarto` and `tinytex` are available, produces a PDF + DOCX; if only `rmarkdown` is available, produces a DOCX. The package never installs LaTeX on its own.

Usage

```
manuscript(run, output_dir = NULL, file = NULL, force = FALSE)
```

Arguments

<code>run</code>	A <code>tx_run</code> object returned by shoot() .
<code>output_dir</code>	Optional output directory.
<code>file</code>	Optional filename stem (without extension).
<code>force</code>	Overwrite an existing file.

Details

Chain stage: **manuscript** (length 2). See [progress\(\)](#).

Value

Character vector of file paths (invisible).

presentation *Generate a chaotic conference-style PPTX*

Description

Builds a multi-slide PowerPoint via `officer`. `v1` produces the structural chaos (rotated text, gradient fills, irrelevant clipart, shape collisions); slide-transition and per-element animation XML injection lands in a follow-up.

Usage

```
presentation(run, output_dir = NULL, file = NULL, force = FALSE)
```

Arguments

<code>run</code>	A <code>tx_run</code> object returned by <code>shoot()</code> .
<code>output_dir</code>	Optional output directory.
<code>file</code>	Optional filename stem (without extension).
<code>force</code>	Overwrite an existing file.

Details

Chain stage: **presentation** (length 3). See `progress()`.

Value

A length-one character vector giving the path to the written PPTX file, returned invisibly. Called for its side effect of generating the slide deck in `output_dir` and advancing the publication chain.

progress *Inspect API unlock progress*

Description

Prints a HUD-style summary of unlock state for gated functions and achievements. Reads live save state.

Usage

```
progress(what = NULL)
```

Arguments

<code>what</code>	Optional. A gated-function name (e.g. "manuscript") or an achievement id (e.g. "ach_multiple_comparisons"). When <code>NULL</code> , prints the overview.
-------------------	---

Details

Three call modes:

- `progress()` – overview: career tier, gated-function lock map, achievement / wardrobe counts, and in-flight progress.
- `progress("manuscript")` – per-function card.
- `progress("ach_multiple_comparisons")` – per-achievement card.

Value

A `tx_progress` object (invisible).

reset	<i>Reset persisted texanshootR state</i>
-------	--

Description

These functions remove pieces of the persistent save written under `tools::R_user_dir()`. They prompt for confirmation in interactive sessions; pass `force = TRUE` to suppress the prompt. Non-interactive sessions require `force = TRUE`.

Usage

```
reset_career(force = FALSE)
```

```
reset_achievements(force = FALSE)
```

```
reset_wardrobe(force = FALSE)
```

```
reset_all(force = FALSE)
```

Arguments

`force` Logical. Skip the confirmation prompt.

Value

TRUE invisibly on success, FALSE if cancelled.

reviewer_response	<i>Generate a response-to-reviewers DOCX</i>
-------------------	--

Description

Polite, point-by-point response to imagined reviewer comments.

Usage

```
reviewer_response(run, output_dir = NULL, file = NULL, force = FALSE)
```

Arguments

run	A tx_run object returned by shoot() .
output_dir	Optional output directory.
file	Optional filename stem (without extension).
force	Overwrite an existing file.

Details

Chain stage: **reviewer_response** (length 4). See [progress\(\)](#).

Value

A length-one character vector giving the path to the written DOCX file, returned invisibly. Called for its side effect of generating the response-to-reviewers document in `output_dir` and advancing the publication chain.

run_log	<i>Browse persisted run history</i>
---------	-------------------------------------

Description

Returns a data.frame of recent runs with one row per run. The full run record can be loaded via `attr(run_log(), "records")[[i]]`.

Usage

```
run_log(n = 25L)
```

Arguments

n	Maximum number of runs to list.
---	---------------------------------

Value

A data.frame with one row per run.

shoot	<i>Run an exploratory model search</i>
-------	--

Description

Fits a battery of candidate specifications across predictor subsets, transformations, interactions, principled-sounding sample restrictions (complete cases, IQR fences, Cook's D, factor-level restrictions), and outcome-engineering moves (composite indices, residualisation, ratios, within-group z-scoring). Returns a `tx_run` object summarising the search and the highlighted specification.

Usage

```
shoot(
  df,
  formula = NULL,
  seed = NULL,
  depth = c("default", "demo"),
  terminal = FALSE,
  interactive_modifiers = FALSE,
  abstract = FALSE,
  manuscript = FALSE,
  presentation = FALSE,
  reviewer_response = FALSE,
  graphical_abstract = FALSE,
  funding = FALSE,
  ...
)
```

Arguments

<code>df</code>	A data frame.
<code>formula</code>	Optional formula. When <code>NULL</code> , the first numeric column is used as the outcome and all other columns as predictors.
<code>seed</code>	Integer seed. When <code>NULL</code> , a random seed is generated and stored on the returned run.
<code>depth</code>	"default" for the full search; "demo" for a single-fit smoke test used in CRAN-safe examples.
<code>terminal</code>	When <code>TRUE</code> , spawn an external Windows Terminal (or fallback console window) and run <code>shoot()</code> inside it so the full ANSI multi-zone TUI renders. Useful when calling from RStudio Console, which only supports <code>\r</code> -overwrite and so falls back to the compressed single-line status. The result is returned to the calling session as usual. Windows-only.
<code>interactive_modifiers</code>	When <code>TRUE</code> , open a <code>readline()</code> tactical-prompt window at each mascot transition so the player can type <code>+command modifiers live</code> . Default <code>FALSE</code> : modifiers

are pre-rolled at run start and applied autonomously, which keeps the in-place animation intact in RStudio's console.

abstract, manuscript, presentation, reviewer_response,
graphical_abstract, funding

Logical flags. When TRUE and the run is shippable, auto-generate the matching output and every chain stage required to reach it. Default FALSE.

...

Reserved for future arguments and aliases. Recognised aliases: powerpoint (= presentation), reviewer (= reviewer_response), graphical (= graphical_abstract). Internal: prompt_fn can be supplied to drive tactical-pause input non-interactively (tests).

Details

Every shoot has a wall-clock budget (default 30s). Modifiers are pre-rolled at run start by default: shoot() picks a random per- transition loadout (+glmm, +derived_metrics, ...) and applies each one when the matching mascot state transition fires, extending the deadline by the modifier's per-token bonus (2-5s) and redirecting the search priority. Pass interactive_modifiers = TRUE to opt into the readline tactical-prompt window at each transition instead. Without +derived_metrics, runs that fail to clear $p \leq 0.05$ simply lose — the highlighted spec stays above threshold and the output gauntlet does not open.

Output flags (abstract, manuscript, presentation / powerpoint, reviewer_response / reviewer, graphical_abstract / graphical, funding) auto-generate the matching file when the run is shippable. The chain prefix needed to reach the highest enabled flag is generated in order: e.g. presentation = TRUE produces abstract, manuscript, and presentation.

Value

A tx_run object. Returned visibly so the print() banner fires at the prompt; the demo path (depth = "demo") returns invisibly because there is no banner worth showing for a one-fit smoke test.

texanshootR-options *Package options for texanshootR*

Description

All options are read at the time of use, so changing them takes effect immediately for subsequent calls.

Details

texanshootR.animations Logical. Enable terminal animations and ANSI cursor effects. Default: TRUE. Forced off in non-interactive sessions and when the terminal lacks ANSI support.

texanshootR.output_dir Character or NULL. Default directory for output files. NULL falls back to `tempdir()`.

texanshootR.save_enabled Logical. Enable persistent career data. FALSE runs entirely in memory. Default: TRUE.

texanshootR.quiet Logical. Suppress status prints from output functions. Default: FALSE.
 texanshootR.event_rate Numeric in [0, 1]. Per-run probability of a life-event encounter. Default: 1/6.
 texanshootR.life_events Logical. Enable life-event encounters during runs. Default: TRUE.
 texanshootR.ui_mode Optional character override for the TUI rendering mode. One of "ansi", "dynamic", "plain". Unset (the default) means auto-detect. Use this if auto-detect picks the wrong mode for your terminal (e.g., a remote RStudio session that does not report `.Platform$GUI == "RStudio"`).

validate_messages	<i>Validate the on-disk message registry</i>
-------------------	--

Description

Lints every YAML file under `inst/messages/` (and an additional path if supplied), enforcing required fields, vocabulary membership, id uniqueness, and combo-chain integrity. Used by the test suite and available to contributors.

Usage

```
validate_messages(path = NULL)
```

Arguments

path	Optional character path to a directory of YAML files. When NULL, validates the package's bundled messages.
------	--

Value

A data.frame of the validated registry, returned invisibly.

wardrobe	<i>Inspect or modify the mascot wardrobe</i>
----------	--

Description

Without arguments, prints the equipped + unlocked cosmetics. With `slot` and `id`, equips the given cosmetic in the given slot.

Usage

```
wardrobe(slot = NULL, id = NULL)
```

Arguments

slot	Optional slot name: hat, poncho, badge, lanyard, cloak.
id	Optional cosmetic id.

Value

The wardrobe state (invisible when modifying, visible when listing).

Index

abstract, [2](#)
achievements, [3](#)

career, [3](#)

funding, [4](#)

graphical_abstract, [4](#)

manuscript, [5](#)

presentation, [6](#)
progress, [6](#)
progress(), [2–6, 8](#)

reset, [7](#)
reset_achievements (reset), [7](#)
reset_all (reset), [7](#)
reset_career (reset), [7](#)
reset_wardrobe (reset), [7](#)
reviewer_response, [8](#)
run_log, [8](#)

shoot, [9](#)
shoot(), [2, 4–6, 8](#)

tempdir(), [10](#)
texanshootR-options, [10](#)
tools::R_user_dir(), [7](#)

validate_messages, [11](#)

wardrobe, [11](#)