

# Package ‘exams.forge.data’

December 18, 2025

**Type** Package

**Title** Sample and Precomputed Data for Use with 'exams.forge'

**Version** 0.1.3

**Description** Provides a small collection of datasets supporting Pearson correlation and linear regression analysis. It includes the precomputed dataset 'sos100', with integer values summing to zero and squared sum equal to 100. For other values of 'n' and user-defined parameters, the 'sos()' function from the 'exams.forge' package can be used to generate datasets on the fly. In addition, the package contains around 500 german R Markdown exercises that illustrate the usage of 'exams.forge' commands.

**License** GPL-3

**Encoding** UTF-8

**LazyData** true

**RoxygenNote** 7.3.3

**Depends** R (>= 3.5)

**Suggests** AER, knitr, rmarkdown

**Imports** exams, exams.forge

**VignetteBuilder** knitr

**NeedsCompilation** no

**Author** Sigbert Klinke [aut, cre] (ORCID:  
<<https://orcid.org/0000-0003-3337-1863>>)

**Maintainer** Sigbert Klinke <sigbert@hu-berlin.de>

**Repository** CRAN

**Date/Publication** 2025-12-18 16:20:07 UTC

## Contents

sos100 . . . . .	2
view . . . . .	2

<b>Index</b>	<b>4</b>
--------------	----------

---

`sos100`*Precomputed Sum of Squared Data*

---

**Description**

A data matrix with precomputed results from `sumofsquares(n, 10, zerosum=TRUE, maxt=Inf)` for  $n=100$ .

More generally, the `sos()` function can be used to retrieve or compute results from `sumofsquares(n, nmax, zerosum=TRUE, maxt=Inf)`.

**Usage**

```
data(sos100)
```

**Format**

For each line of the matrix it holds  $\sum_{i=1}^k x_i^2 = n$  and  $\sum_{i=1}^k x_i = 0$ . It contains all integer solutions up to  $k \leq 10$ . NA means that this entry is not used.

**Examples**

```
data(sos100)
head(sos100)
rowSums(sos100^2, na.rm=TRUE)
rowSums(sos100, na.rm=TRUE)
```

---

`view`*Explore and Render Exam Exercises*

---

**Description**

This function allows the user to explore exercises from the `exams.forge.data` package and optionally render them to PDF or HTML. Users can filter exercises by directory (topic) or by a pattern in filenames or file contents.

**Usage**

```
view(
  action = c("topic", "file", "pdf", "html"),
  pattern = NULL,
  topic = TRUE,
  ...
)
```

**Arguments**

action	Character. One of "topic", "file", "pdf", or "html". <ul style="list-style-type: none"><li>• "topic": return the list of available exercise topics (subdirectories).</li><li>• "file": return the list of exercise filenames relative to aufgaben/.</li><li>• "pdf": render selected exercises to PDF.</li><li>• "html": render selected exercises to HTML.</li></ul>
pattern	Character or NULL. A regular expression to filter exercises. If topic = TRUE, the pattern is matched against the directory names (topics). If topic = FALSE, the pattern is searched inside the exercise files.
topic	Logical. If TRUE (default), pattern is applied to directory names. If FALSE, pattern is applied to the content of the exercise files.
...	Additional arguments passed to <code>grep1()</code> when filtering files or contents.

**Value**

A character vector of topics or filenames, or NULL if no exercises are found. When rendering (pdf/html), the files are rendered silently with messages and warnings suppressed.

**Examples**

```
view("topic")

# List all files in a specific topic
view("file", pattern = "ttest", topic = TRUE)

if (interactive()) {
  # Render the files matching a pattern to HTML
  view("file", pattern = "hyper")
  view("html", pattern = "hyper")
}
```

# Index

\* **datasets**

[sos100, 2](#)

[sos100, 2](#)

[view, 2](#)