

# Package: zipper (via r-universe)

November 6, 2024

**Type** Package

**Maintainer** Steven E. Pav <shabbychef@gmail.com>

**Version** 0.1.0.1000

**Date** 2017-05-03

**License** LGPL-3

**Title** Zip Sorted Arrays Together

**BugReports** <https://github.com/shabbychef/zipper/issues>

**Description** Just zips together sorted arrays.

**Imports** Rcpp (>= 0.12.3)

**LinkingTo** Rcpp

**Suggests** testthat

**RoxygenNote** 5.0.1

**URL** <https://github.com/shabbychef/zipper>

**Collate** 'RcppExports.R' 'zipper.r'

**Repository** <https://shabbychef.r-universe.dev>

**RemoteUrl** <https://github.com/shabbychef/zipper>

**RemoteRef** HEAD

**RemoteSha** c79b5cce59b1fc64c4348274a3b443294af00173

## Contents

zipper-package . . . . .	2
zipper-NEWS . . . . .	2
zipsorted . . . . .	2

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

---

zipper-package	<i>Zip Sorted Arrays Together</i>
----------------	-----------------------------------

---

**Description**

That's all it does. zips them.

**Legal Mumbo Jumbo**

zipper is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

**Author(s)**

Steven E. Pav <shabbychef@gmail.com>

---

zipper-NEWS	<i>News for package 'zipper':</i>
-------------	-----------------------------------

---

**Description**

News for package 'zipper'

**Initial Version 0.1.0 (2017-05-01)**

- start work

---

zipsorted	<i>Zip sorted arrays against each other</i>
-----------	---

---

**Description**

Given two sorted arrays,  $x$  and  $y$ , find indices  $L$  that 'zips' the two together.

**Usage**

```
zip_le(sortx, looky)
```

```
zip_lt(sortx, looky)
```

**Arguments**

sortx	a sorted array of 'reference' values.
looky	a sorted array of values whose place among sortx is to be found.

**Details**

For example, for `zip_le`, we find the array  $L$  of the same length as  $y$  such that there are exactly  $L_i$  elements of  $x$  less than or equal to  $y_i$ .

**Value**

a vector, filled out as follows:

**zip\_le** Returns the vector  $L$  such that there are exactly  $L_i$  elements of  $x$  less than or equal to  $y_i$ .

**zip\_lt** Returns the vector  $L$  such that there are exactly  $L_i$  elements of  $x$  less than  $y_i$ .

**Note**

Returns zero when there are none, as expected.

it would be better if this code supported mixed types of `sortx` and `looky`.

**Author(s)**

Steven E. Pav <shabbychef@gmail.com>

**Examples**

```
set.seed(1234)
x <- sort(rnorm(1e5))
y <- sort(rnorm(1e2))
idx1 <- zip_le(x,y)
# slow way, should give the same answer
uthr <- rep(NA,length(y))
for (iii in 1:length(y)) {
  uthr[iii] <- sum(x <= y[iii])
}
```

# Index

## \* **package**

zipper-package, [2](#)

zip\_le (zipsorted), [2](#)

zip\_lt (zipsorted), [2](#)

zipper-NEWS, [2](#)

zipper-package, [2](#)

zipsorted, [2](#)