

Package ‘bitsqueezr’

January 17, 2020

Type Package

Title Quantize Floating-Point Numbers for Improved Compressibility

Version 0.1.1

Author Daniel Baston <dbaston@gmail.com>

Maintainer Daniel Baston <dbaston@gmail.com>

Description Provides a implementation of floating-point quantization algorithms for use in precision-preserving compression, similar to the approach taken in the 'netCDF operators' (NCO) software package and described in Zender (2016) <doi:10.5194/gmd-2016-63>.

License GPL-3

URL <https://github.com/dbaston/bitsqueezr>

BugReports <https://github.com/dbaston/bitsqueezr/issues>

Encoding UTF-8

LazyData true

Suggests testthat, covr

NeedsCompilation yes

Repository CRAN

Date/Publication 2020-01-17 15:30:02 UTC

R topics documented:

bits_as_string	2
squeeze_bits	2

Index	4
--------------	----------

bits_as_string	<i>Generate a text representation of the bits in a double</i>
----------------	---

Description

Generate a character vector of ones, zeros, and spaces, representing the bits in a double. Spaces are used to separate the sign bit, 11 exponent bits, and 52 fraction bits.

Usage

```
bits_as_string(d)
```

Arguments

d	a numeric value
---	-----------------

Examples

```
# Output binary representation of pi
bits_as_string(pi)
# 0 10000000000 1001001000011111101101010100010001000010110100011000
```

squeeze_bits	<i>Change insignificant bits of numeric values for improved compressibility</i>
--------------	---

Description

Change insignificant bits of numeric values to zero or one, increasing the compressibility of files containing the values. Insignificant bits can be "trimmed" (set to zero), "padded" (set to one), or "groomed" (element-wise alternation between trimming and padding). A discussion of these schemes is provided by Zender, Charles (2016) *Statistically-accurate precision-preserving quantization with compression, evaluated in the netCDF operators. Geoscientific Model Development* 9(9). The file size reduction depends on the level of quantization and the compression algorithm used.

Usage

```
squeeze_bits(x, digits, method = 'trim', decimal = FALSE)
```

Arguments

x	a numeric vector
digits	number of digits to preserve
method	'trim' sets insignificant bits to zero, 'pad' sets insignificant bits to one, and 'groom' alternates between 'trim' and 'pad'
decimal	if TRUE, d will be interpreted to refer to decimal digits rather than significant digits.

Examples

```
# Check file size reduction when retaining 6 significant digits
x <- runif(100)
raw <- tempfile(fileext='.rds')
quantized <- tempfile(fileext='.rds')

saveRDS(x, raw, compress='xz')
saveRDS(squeeze_bits(x, 6, method='trim'), quantized, compress='xz')

file.size(quantized) / file.size(raw)
# 0.6776316

# Display binary representation of pi with various levels of trimming
for (d in 1:15) {
  cat(bits_as_string(squeeze_bits(pi, d, method='trim')), '\n')
}
```

Index

`bits_as_string`, [2](#)

`squeeze_bits`, [2](#)