Package ‘schoenberg’

January 28, 2024

Type Package
Title Tools for 12-Tone Musical Composition
Version 2.0.3
Date 2024-01-27
Author Jeffrey A. Dahlke <jeff.dahlke.phd@gmail.com>
Maintainer Jeffrey A. Dahlke <jeff.dahlke.phd@gmail.com>
Description Functions for creating and manipulating 12-tone (i.e., dodecaphonic) musical matrices using Arnold Schoenberg's (1923) serialism technique. This package can generate random 12-tone matrices and can generate matrices using a pre-determined sequence of notes.

BugReports https://github.com/jadahlke/schoenberg/issues
License GPL (>= 3)
Imports crayon
Encoding UTF-8
RoxygenNote 7.2.3
NeedsCompilation no
Repository CRAN
Date/Publication 2024-01-28 11:20:03 UTC

R topics documented:

  print .......................................................... 2
  rekey .......................................................... 2
  schoenberg ................................................. 3

Index 6
**print**

*Print methods for schoenberg*

**Description**

Print methods for *schoenberg* output objects with classes exported from *schoenberg*.

**Arguments**

- **x**: Object to be printed (object is used to select a method).
- **...**: Additional arguments.

---

**rekey**

*Re-express a "schoenberg" class object with a different lead tone or different notation of accidentals.*

**Description**

Re-express a "schoenberg" class object with a different lead tone or different notation of accidentals.

**Usage**

```r
tone_mat, tone0 = NULL, accidentals = NULL)
```

**Arguments**

- **tone_mat**: Object of the class "schoenberg" produced by the *schoenberg()* function.
- **tone0**: `Optional`: Name of the note to use as the lead tone of the matrix.
- **accidentals**: `Optional`: Character scalar that determines whether accidentals should be represented as sharps (accidentals = "sharps") or flats (accidentals = "flats"); default value is NULL. accidentals can also be set to "integers" when one wishes to obtain a 12-tone matrix of numeric indices rather than notes. When accidentals is NULL, matrices created from pre-specified vectors of notes will use the original set of accidentals, whereas random matrices and matrices created from vectors of numeric indices will default to sharp notation.

**Value**

A 12-tone matrix of the "schoenberg" class with prime series on the rows and inverted series on the columns.
Examples

```r
# Let's create a vector of notes to use in creating our initial 'tone_mat' matrix based
# on Schoenberg's Walzer from Opus 23
tone_mat <- schoenberg(prime0 = prime01)

# Now, let's change the lead tone to "C":
rekey(tone_mat = tone_mat, tone0 = "C")

# And let's also change the accidentals to flats:
rekey(tone_mat = tone_mat, tone0 = "C", accidentals = "flats")
```

---

`schoenberg`  
Generate a 12-tone matrix using Arnold Schoenberg’s serialism technique.

**Description**
Generate a 12-tone matrix using Arnold Schoenberg’s serialism technique.

**Usage**

`schoenberg(prime0 = NULL, tone0 = NULL, accidentals = NULL, seed = NULL)`

**Arguments**

- **prime0**  
  *Optional*: Vector of notes or numeric note indices to use in forming the matrix. If the vector is numeric, the values must span from 0 - 11, where 0 is the lead tone (unless `tone0` is specified, note 0 will be treated as "C"). If supplying note names, use capital letters for the note names, use "#" to indicate sharps, and use "b" to indicate flats.

- **tone0**  
  *Optional*: Name of the note to use as the lead tone of the matrix.

- **accidentals**  
  *Optional*: Character scalar that determines whether accidentals should be represented as sharps (`accidentals = "sharps"`) or flats (`accidentals = "flats"`); default value is `NULL`. `accidentals` can also be set to "integers" when one wishes to obtain a 12-tone matrix of numeric indices rather than notes. When `accidentals` is `NULL`, matrices created from pre-specified vectors of notes will use the original set of accidentals, whereas random matrices and matrices created from vectors of numeric indices will default to sharp notation.

- **seed**  
  *Optional*: Seed value to use in generating random matrices. Set this to a numeric value when matrices need to be reproducible.

**Value**
A 12-tone matrix of the "schoenberg" class with prime series on the rows and inverted series on the columns.
References


Examples

#### Generating Random 12-Tone Matrices ####
# The schoenberg() function can generate completely random 12-tone matrices:
schoenberg()

# Or you can specify a seed value so that your matrices are reproducible:
schoenberg(seed = 42)

#### Generating 12-Tone Matrices From a Specified Vector of Notes ####
# For illustration, let's create two equivalent vectors of note information
# for Schoenberg's first 12-tone serialist work: Walzer from Opus 23.

# First, let's create one vector with note labels:

# Next, let's create an equivalent vector using numeric indices instead of notes:
prime02 <- c(1, 9, 11, 7, 8, 6, 10, 2, 4, 3, 0, 5)

# Now, let's generate a 12-tone matrix from our note-based vector:
schoenberg(prime0 = prime01)

# And let's generate a matrix from our number-based vector:
schoenberg(prime0 = prime02)

# Schoenberg used a mix of sharps and flats in his notation, which lost in translation with the
# numeric-index approach. Let's re-create our note-based matrix using only sharps:
schoenberg(prime0 = prime01, accidentals = "sharps")

# These two approaches produce identical outputs:
all(schoenberg(prime0 = prime01, accidentals = "sharps") == schoenberg(prime0 = prime02))

# Matrices can also be generated with flat notation by setting accidentals to "flats":
schoenberg(prime0 = prime01, accidentals = "flats")
schoenberg(prime0 = prime02, accidentals = "flats")

# As before, these two approaches produce identical outputs:
all(schoenberg(prime0 = prime01, accidentals = "flats") ==
    schoenberg(prime0 = prime02, accidentals = "flats"))

# We can also manipulate the output of the schoenberg() function
# so that the lead tone of the matrix is a particular note.
# This works with either note-based or number-based input vectors:
schoenberg(prime0 = prime01, tone0 = "C", accidentals = "sharps")
schoenberg(prime0 = prime02, tone0 = "C")

# And, as before, these two approaches produce identical outputs:
all(schoenberg(prime0 = prime01, tone0 = "C", accidentals = "sharps") ==
    schoenberg(prime0 = prime02, tone0 = "C"))
Index

print, 2
rekey, 2
schoenberg, 3