



Working with Missing, Approximate, Uncertain, Sets and Ranges of Dates with **messydates**

James Hollway

Graduate Institute of

International and Development Studies

Henrique Sposito

Graduate Institute of

International and Development Studies

Abstract

This paper presents the **messydates** package for R, which facilitates working with ‘messy’ dates. Messy dates are common when studying historical and sometimes even current phenomena, and can create various technical problems for the data analyst. The paper highlights these problems and offers practical advice on how to solve them using **messydates**. The paper also introduces a conceptual framework for resolving messydates into more familiar date classes in R ready for analysis.

Keywords: dates, ISO, R.

1. Introduction

Dates are often messy. Whether historical (or ancient), future, or even recent, we often only know approximately when an event occurred, that it happened within a particular period, an unreliable source means a date should be flagged as uncertain, or sources offer multiple, competing dates.

messydates implements the extended annotation standard for dates, the Extended Date/Time Format (EDTF), outlined in [ISO 8601-2_2019\(E\)](#) for R. These include standardised notation for:

- unspecified date(component)s, e.g. 2012-XX-01 for the first of some unknown month in 2012 or 2012-01 for some unknown day in January 2012
- approximate date(component)s, e.g. 2012-01-12~ for approximately the 12th of January 2012
- uncertain date(component)s, e.g. 2012-01-12? where this data point is based on an unreliable source

- sets of dates, e.g. `{2012-01-01,2012-01-12}` where the date can be both 1 January 2012 and 12 January 2012
- ranges of dates, e.g. `2012-01-01..2012-01-12` for all dates between the 1 January 2012 and 12 January 2012 inclusive

messydates contains a set of tools for constructing and coercing into and from the `mdate` class. This date class allows regular dates to be annotated to express unspecified date components, approximate or uncertain date components, date ranges, and sets of dates.

Importantly, the package also includes a function for unpacking or expanding sets or ranges of dates into all dates consistent with how the date or set of dates is specified or annotated. Methods are also offered that can be used to make explicit how researchers convert date imprecision into precise dates for analysis, such as getting the `min()`, `max()`, or even a `random()` date from among the dates consistent with a set or range of dates. This greatly facilitates research transparency as well as robustness checks.

1.1. Motivation

As researchers, we often recognize this messiness but are forced to force non-existent precision on data so we can proceed with analysis. For example, if we only know something happened in a given month or year, we might just opt for the start of that month (e.g. `2021-07-01`) or year (`2021-01-01`), assuming that to err on the earlier (or later) side is a justifiable bias. However, this can create issues for inference in which sequence or timing is important. The goal of **messydates** is to help with this problem by retaining and working with various kinds of date imprecision.

1.2. Relationship to other packages

messydates offers a new date class, but one that comes with methods for converting from and into common date classes such as `Date`, `POSIXct`, and `POSIXlt`. It is thus fully compatible with packages such as **lubridate** (Grolemund and Wickham 2011) and **anytime** (Eddelbuettel 2019).

2. R code

2.1. A new class

messydates contains a set of tools for constructing and coercing into and from the `mdate` class. This date class implements ISO 8601-2:2019(E) and allows regular dates to be annotated to express unspecified date components, approximate or uncertain date components, date ranges, and sets of dates. The function `as_messydate()` handles the coercion to `mdate` class.

```
R> library(messydates)
R> suppressPackageStartupMessages(library(lubridate))
R> suppressPackageStartupMessages(library(anytime))
R> library(tibble)
R> suppressPackageStartupMessages(library(dplyr))
```

```
R> dates_comparison <- tibble::tribble(~Example, ~OriginalDate,
+                                     "A normal date", as.character(Sys.Date()),
+                                     "A future date", "2599-12-31",
+                                     "A written date", "First of February, two thousand a",
+                                     "A historical date", "476",
+                                     "An era date", "33 BC",
+                                     "An approximate date", "2012-01-12~",
+                                     "An uncertain date", "2001-01-01?",
+                                     "An unspecified date", "2012-01",
+                                     "A censored date", "..2012-01-12",
+                                     "A range of dates", "2019-11-01:2020-01-01",
+                                     "A set of dates", "2021-5-26, 2021-11-19, 2021-12-4",
+   dplyr::mutate(base = as.Date(OriginalDate),
+                 lubridate = lubridate::as_date(OriginalDate),
+                 anytime = anytime::anydate(OriginalDate),
+                 messydates = messydates::as_messydate(OriginalDate)) %>%
+   print())
```

```
# A tibble: 11 x 6
```

	Example <chr>	OriginalDate <chr>	base <date>	lubridate <date>	anytime <date>	messydates <mdate>
1	A normal date	2022-06-15	2022-06-15	2022-06-15	2022-06-15	2022-06-1~
2	A future date	2599-12-31	2599-12-31	2599-12-31	2599-12-31	2599-12-3~
3	A written date	First of Feb~	NA	NA	NA	2021-02-0~
4	A historical date	476	NA	NA	NA	0476 ~
5	An era date	33 BC	NA	NA	NA	-0033 ~
6	An approximate date	2012-01-12~	2012-01-12	2012-01-12	2012-01-12	2012-01-1~
7	An uncertain date	2001-01-01?	2001-01-01	2001-01-01	2001-01-01	2001-01-0~
8	An unspecified date	2012-01	NA	2020-12-01	2012-01-01	2012-01 ~
9	A censored date	..2012-01-12	NA	2012-01-12	NA	..2012-01~
10	A range of dates	2019-11-01:2~	2019-11-01	2019-11-01	2019-11-01	2019-11-0~
11	A set of dates	2021-5-26, 2~	2021-05-26	NA	2021-05-26	{2021-05--

2.2. Annotate

Some datasets have, for example, an arbitrary cut off point for start and end points, but these are often coded as precise dates when they are not necessarily the real start or end dates. The `annotate` functions helps annotate uncertainty and approximation to dates. Inaccurate start or end dates can be represented by an affix indicating “on or before”, if used as a prefix (e.g. `..1816-01-01`), or indicating “on or after”, if used as a suffix (e.g. `2016-12-31..`). Approximate dates are indicated by adding a `~` to year, month, or day components, as well as groups of components or whole dates to estimate values that are possibly correct (e.g. `2003-03-03~`). Day, month, or year, uncertainty can be indicated by adding a `?` to a possibly dubious date (e.g. `1916-10-10?`) or date component (e.g. `1916-?10-10`).

```
R> dates_annotate <- tibble::tibble(Beg = as_messydate(c("1816-01-01", "1916-01-01", "2016-01-01")))
```

```

+                               End = as_messydate(c("1816-12-31", "1916-12-31", "2016-
R> dplyr::mutate(dates_annotate, Beg = ifelse(Beg <= "1816-01-01", on_or_before(Beg), Beg))

# A tibble: 3 x 2
  Beg      End
  <chr>    <mdate>
1 1816-01-01 1816-12-31
2 1916-01-01 1916-12-31
3 2016-01-01 2016-12-31

R> dplyr::mutate(dates_annotate, End = ifelse(End >= "2016-01-01", on_or_after(End), End))

# A tibble: 3 x 2
  Beg      End
  <mdate>    <chr>
1 1816-01-01 1816-12-31
2 1916-01-01 1916-12-31
3 2016-01-01 2016-12-31..

R> dplyr::mutate(dates_annotate, Beg = ifelse(Beg == "1916-01-01", as_approximate(Beg), Beg))

# A tibble: 3 x 2
  Beg      End
  <chr>    <mdate>
1 1816-01-01 1816-12-31
2 1916-01-01~ 1916-12-31
3 2016-01-01 2016-12-31

R> dplyr::mutate(dates_annotate, End = ifelse(End == "1916-12-31", as_uncertain(End), End))

# A tibble: 3 x 2
  Beg      End
  <mdate>    <chr>
1 1816-01-01 1816-12-31
2 1916-01-01 1916-12-31?
3 2016-01-01 2016-12-31

```

2.3. Expand

Expand functions transform date ranges, sets of dates, and unspecified or approximate dates (annotated with ‘.’, ‘{ , }’, ‘XX’ or ‘~’) into lists of dates. As these dates may refer to several possible dates, the function “opens” these values to include all the possible dates implied.

```

R> dates_expand <- as_messydate(c("2001-01-01", "2001-01?", "2001",
+                               "2001-01-01..2001-02-02", "{2001-01-01,2001-02-02}",
+                               "2001-XX-31", "21 BC"))
R> expand(dates_expand)

```

Please specify 'approx_range' argument if you want approximate dates to also be expanded

```
[[1]]
```

```
[1] "2001-01-01"
```

```
[[2]]
```

```
[1] "2001-01-01" "2001-01-02" "2001-01-03" "2001-01-04" "2001-01-05"
[6] "2001-01-06" "2001-01-07" "2001-01-08" "2001-01-09" "2001-01-10"
[11] "2001-01-11" "2001-01-12" "2001-01-13" "2001-01-14" "2001-01-15"
[16] "2001-01-16" "2001-01-17" "2001-01-18" "2001-01-19" "2001-01-20"
[21] "2001-01-21" "2001-01-22" "2001-01-23" "2001-01-24" "2001-01-25"
[26] "2001-01-26" "2001-01-27" "2001-01-28" "2001-01-29" "2001-01-30"
[31] "2001-01-31"
```

```
[[3]]
```

```
[1] "2001-01-01" "2001-01-02" "2001-01-03" "2001-01-04" "2001-01-05"
[6] "2001-01-06" "2001-01-07" "2001-01-08" "2001-01-09" "2001-01-10"
[11] "2001-01-11" "2001-01-12" "2001-01-13" "2001-01-14" "2001-01-15"
[16] "2001-01-16" "2001-01-17" "2001-01-18" "2001-01-19" "2001-01-20"
[21] "2001-01-21" "2001-01-22" "2001-01-23" "2001-01-24" "2001-01-25"
[26] "2001-01-26" "2001-01-27" "2001-01-28" "2001-01-29" "2001-01-30"
[31] "2001-01-31" "2001-02-01" "2001-02-02" "2001-02-03" "2001-02-04"
[36] "2001-02-05" "2001-02-06" "2001-02-07" "2001-02-08" "2001-02-09"
[41] "2001-02-10" "2001-02-11" "2001-02-12" "2001-02-13" "2001-02-14"
[46] "2001-02-15" "2001-02-16" "2001-02-17" "2001-02-18" "2001-02-19"
[51] "2001-02-20" "2001-02-21" "2001-02-22" "2001-02-23" "2001-02-24"
[56] "2001-02-25" "2001-02-26" "2001-02-27" "2001-02-28" "2001-03-01"
[61] "2001-03-02" "2001-03-03" "2001-03-04" "2001-03-05" "2001-03-06"
[66] "2001-03-07" "2001-03-08" "2001-03-09" "2001-03-10" "2001-03-11"
[71] "2001-03-12" "2001-03-13" "2001-03-14" "2001-03-15" "2001-03-16"
[76] "2001-03-17" "2001-03-18" "2001-03-19" "2001-03-20" "2001-03-21"
[81] "2001-03-22" "2001-03-23" "2001-03-24" "2001-03-25" "2001-03-26"
[86] "2001-03-27" "2001-03-28" "2001-03-29" "2001-03-30" "2001-03-31"
[91] "2001-04-01" "2001-04-02" "2001-04-03" "2001-04-04" "2001-04-05"
[96] "2001-04-06" "2001-04-07" "2001-04-08" "2001-04-09" "2001-04-10"
[101] "2001-04-11" "2001-04-12" "2001-04-13" "2001-04-14" "2001-04-15"
[106] "2001-04-16" "2001-04-17" "2001-04-18" "2001-04-19" "2001-04-20"
[111] "2001-04-21" "2001-04-22" "2001-04-23" "2001-04-24" "2001-04-25"
[116] "2001-04-26" "2001-04-27" "2001-04-28" "2001-04-29" "2001-04-30"
[121] "2001-05-01" "2001-05-02" "2001-05-03" "2001-05-04" "2001-05-05"
[126] "2001-05-06" "2001-05-07" "2001-05-08" "2001-05-09" "2001-05-10"
[131] "2001-05-11" "2001-05-12" "2001-05-13" "2001-05-14" "2001-05-15"
[136] "2001-05-16" "2001-05-17" "2001-05-18" "2001-05-19" "2001-05-20"
[141] "2001-05-21" "2001-05-22" "2001-05-23" "2001-05-24" "2001-05-25"
[146] "2001-05-26" "2001-05-27" "2001-05-28" "2001-05-29" "2001-05-30"
[151] "2001-05-31" "2001-06-01" "2001-06-02" "2001-06-03" "2001-06-04"
[156] "2001-06-05" "2001-06-06" "2001-06-07" "2001-06-08" "2001-06-09"
```

```

[161] "2001-06-10" "2001-06-11" "2001-06-12" "2001-06-13" "2001-06-14"
[166] "2001-06-15" "2001-06-16" "2001-06-17" "2001-06-18" "2001-06-19"
[171] "2001-06-20" "2001-06-21" "2001-06-22" "2001-06-23" "2001-06-24"
[176] "2001-06-25" "2001-06-26" "2001-06-27" "2001-06-28" "2001-06-29"
[181] "2001-06-30" "2001-07-01" "2001-07-02" "2001-07-03" "2001-07-04"
[186] "2001-07-05" "2001-07-06" "2001-07-07" "2001-07-08" "2001-07-09"
[191] "2001-07-10" "2001-07-11" "2001-07-12" "2001-07-13" "2001-07-14"
[196] "2001-07-15" "2001-07-16" "2001-07-17" "2001-07-18" "2001-07-19"
[201] "2001-07-20" "2001-07-21" "2001-07-22" "2001-07-23" "2001-07-24"
[206] "2001-07-25" "2001-07-26" "2001-07-27" "2001-07-28" "2001-07-29"
[211] "2001-07-30" "2001-07-31" "2001-08-01" "2001-08-02" "2001-08-03"
[216] "2001-08-04" "2001-08-05" "2001-08-06" "2001-08-07" "2001-08-08"
[221] "2001-08-09" "2001-08-10" "2001-08-11" "2001-08-12" "2001-08-13"
[226] "2001-08-14" "2001-08-15" "2001-08-16" "2001-08-17" "2001-08-18"
[231] "2001-08-19" "2001-08-20" "2001-08-21" "2001-08-22" "2001-08-23"
[236] "2001-08-24" "2001-08-25" "2001-08-26" "2001-08-27" "2001-08-28"
[241] "2001-08-29" "2001-08-30" "2001-08-31" "2001-09-01" "2001-09-02"
[246] "2001-09-03" "2001-09-04" "2001-09-05" "2001-09-06" "2001-09-07"
[251] "2001-09-08" "2001-09-09" "2001-09-10" "2001-09-11" "2001-09-12"
[256] "2001-09-13" "2001-09-14" "2001-09-15" "2001-09-16" "2001-09-17"
[261] "2001-09-18" "2001-09-19" "2001-09-20" "2001-09-21" "2001-09-22"
[266] "2001-09-23" "2001-09-24" "2001-09-25" "2001-09-26" "2001-09-27"
[271] "2001-09-28" "2001-09-29" "2001-09-30" "2001-10-01" "2001-10-02"
[276] "2001-10-03" "2001-10-04" "2001-10-05" "2001-10-06" "2001-10-07"
[281] "2001-10-08" "2001-10-09" "2001-10-10" "2001-10-11" "2001-10-12"
[286] "2001-10-13" "2001-10-14" "2001-10-15" "2001-10-16" "2001-10-17"
[291] "2001-10-18" "2001-10-19" "2001-10-20" "2001-10-21" "2001-10-22"
[296] "2001-10-23" "2001-10-24" "2001-10-25" "2001-10-26" "2001-10-27"
[301] "2001-10-28" "2001-10-29" "2001-10-30" "2001-10-31" "2001-11-01"
[306] "2001-11-02" "2001-11-03" "2001-11-04" "2001-11-05" "2001-11-06"
[311] "2001-11-07" "2001-11-08" "2001-11-09" "2001-11-10" "2001-11-11"
[316] "2001-11-12" "2001-11-13" "2001-11-14" "2001-11-15" "2001-11-16"
[321] "2001-11-17" "2001-11-18" "2001-11-19" "2001-11-20" "2001-11-21"
[326] "2001-11-22" "2001-11-23" "2001-11-24" "2001-11-25" "2001-11-26"
[331] "2001-11-27" "2001-11-28" "2001-11-29" "2001-11-30" "2001-12-01"
[336] "2001-12-02" "2001-12-03" "2001-12-04" "2001-12-05" "2001-12-06"
[341] "2001-12-07" "2001-12-08" "2001-12-09" "2001-12-10" "2001-12-11"
[346] "2001-12-12" "2001-12-13" "2001-12-14" "2001-12-15" "2001-12-16"
[351] "2001-12-17" "2001-12-18" "2001-12-19" "2001-12-20" "2001-12-21"
[356] "2001-12-22" "2001-12-23" "2001-12-24" "2001-12-25" "2001-12-26"
[361] "2001-12-27" "2001-12-28" "2001-12-29" "2001-12-30" "2001-12-31"

[[4]]
  [1] "2001-01-01" "2001-01-02" "2001-01-03" "2001-01-04" "2001-01-05"
  [6] "2001-01-06" "2001-01-07" "2001-01-08" "2001-01-09" "2001-01-10"
 [11] "2001-01-11" "2001-01-12" "2001-01-13" "2001-01-14" "2001-01-15"
 [16] "2001-01-16" "2001-01-17" "2001-01-18" "2001-01-19" "2001-01-20"

```

```
[21] "2001-01-21" "2001-01-22" "2001-01-23" "2001-01-24" "2001-01-25"
[26] "2001-01-26" "2001-01-27" "2001-01-28" "2001-01-29" "2001-01-30"
[31] "2001-01-31" "2001-02-01" "2001-02-02"
```

```
[[5]]
```

```
[1] "2001-01-01" "2001-02-02"
```

```
[[6]]
```

```
[1] "2001-01-31" "2001-02-28" "2001-03-31" "2001-04-30" "2001-05-31"
```

```
[6] "2001-06-30" "2001-07-31" "2001-08-31" "2001-09-30" "2001-10-31"
```

```
[11] "2001-11-30" "2001-12-31"
```

```
[[7]]
```

```
[1] "-0021-01-01" "-0021-01-02" "-0021-01-03" "-0021-01-04" "-0021-01-05"
```

```
[6] "-0021-01-06" "-0021-01-07" "-0021-01-08" "-0021-01-09" "-0021-01-10"
```

```
[11] "-0021-01-11" "-0021-01-12" "-0021-01-13" "-0021-01-14" "-0021-01-15"
```

```
[16] "-0021-01-16" "-0021-01-17" "-0021-01-18" "-0021-01-19" "-0021-01-20"
```

```
[21] "-0021-01-21" "-0021-01-22" "-0021-01-23" "-0021-01-24" "-0021-01-25"
```

```
[26] "-0021-01-26" "-0021-01-27" "-0021-01-28" "-0021-01-29" "-0021-01-30"
```

```
[31] "-0021-01-31" "-0021-02-01" "-0021-02-02" "-0021-02-03" "-0021-02-04"
```

```
[36] "-0021-02-05" "-0021-02-06" "-0021-02-07" "-0021-02-08" "-0021-02-09"
```

```
[41] "-0021-02-10" "-0021-02-11" "-0021-02-12" "-0021-02-13" "-0021-02-14"
```

```
[46] "-0021-02-15" "-0021-02-16" "-0021-02-17" "-0021-02-18" "-0021-02-19"
```

```
[51] "-0021-02-20" "-0021-02-21" "-0021-02-22" "-0021-02-23" "-0021-02-24"
```

```
[56] "-0021-02-25" "-0021-02-26" "-0021-02-27" "-0021-02-28" "-0021-03-01"
```

```
[61] "-0021-03-02" "-0021-03-03" "-0021-03-04" "-0021-03-05" "-0021-03-06"
```

```
[66] "-0021-03-07" "-0021-03-08" "-0021-03-09" "-0021-03-10" "-0021-03-11"
```

```
[71] "-0021-03-12" "-0021-03-13" "-0021-03-14" "-0021-03-15" "-0021-03-16"
```

```
[76] "-0021-03-17" "-0021-03-18" "-0021-03-19" "-0021-03-20" "-0021-03-21"
```

```
[81] "-0021-03-22" "-0021-03-23" "-0021-03-24" "-0021-03-25" "-0021-03-26"
```

```
[86] "-0021-03-27" "-0021-03-28" "-0021-03-29" "-0021-03-30" "-0021-03-31"
```

```
[91] "-0021-04-01" "-0021-04-02" "-0021-04-03" "-0021-04-04" "-0021-04-05"
```

```
[96] "-0021-04-06" "-0021-04-07" "-0021-04-08" "-0021-04-09" "-0021-04-10"
```

```
[101] "-0021-04-11" "-0021-04-12" "-0021-04-13" "-0021-04-14" "-0021-04-15"
```

```
[106] "-0021-04-16" "-0021-04-17" "-0021-04-18" "-0021-04-19" "-0021-04-20"
```

```
[111] "-0021-04-21" "-0021-04-22" "-0021-04-23" "-0021-04-24" "-0021-04-25"
```

```
[116] "-0021-04-26" "-0021-04-27" "-0021-04-28" "-0021-04-29" "-0021-04-30"
```

```
[121] "-0021-05-01" "-0021-05-02" "-0021-05-03" "-0021-05-04" "-0021-05-05"
```

```
[126] "-0021-05-06" "-0021-05-07" "-0021-05-08" "-0021-05-09" "-0021-05-10"
```

```
[131] "-0021-05-11" "-0021-05-12" "-0021-05-13" "-0021-05-14" "-0021-05-15"
```

```
[136] "-0021-05-16" "-0021-05-17" "-0021-05-18" "-0021-05-19" "-0021-05-20"
```

```
[141] "-0021-05-21" "-0021-05-22" "-0021-05-23" "-0021-05-24" "-0021-05-25"
```

```
[146] "-0021-05-26" "-0021-05-27" "-0021-05-28" "-0021-05-29" "-0021-05-30"
```

```
[151] "-0021-05-31" "-0021-06-01" "-0021-06-02" "-0021-06-03" "-0021-06-04"
```

```
[156] "-0021-06-05" "-0021-06-06" "-0021-06-07" "-0021-06-08" "-0021-06-09"
```

```
[161] "-0021-06-10" "-0021-06-11" "-0021-06-12" "-0021-06-13" "-0021-06-14"
```

```
[166] "-0021-06-15" "-0021-06-16" "-0021-06-17" "-0021-06-18" "-0021-06-19"
```

```

[171] "-0021-06-20" "-0021-06-21" "-0021-06-22" "-0021-06-23" "-0021-06-24"
[176] "-0021-06-25" "-0021-06-26" "-0021-06-27" "-0021-06-28" "-0021-06-29"
[181] "-0021-06-30" "-0021-07-01" "-0021-07-02" "-0021-07-03" "-0021-07-04"
[186] "-0021-07-05" "-0021-07-06" "-0021-07-07" "-0021-07-08" "-0021-07-09"
[191] "-0021-07-10" "-0021-07-11" "-0021-07-12" "-0021-07-13" "-0021-07-14"
[196] "-0021-07-15" "-0021-07-16" "-0021-07-17" "-0021-07-18" "-0021-07-19"
[201] "-0021-07-20" "-0021-07-21" "-0021-07-22" "-0021-07-23" "-0021-07-24"
[206] "-0021-07-25" "-0021-07-26" "-0021-07-27" "-0021-07-28" "-0021-07-29"
[211] "-0021-07-30" "-0021-07-31" "-0021-08-01" "-0021-08-02" "-0021-08-03"
[216] "-0021-08-04" "-0021-08-05" "-0021-08-06" "-0021-08-07" "-0021-08-08"
[221] "-0021-08-09" "-0021-08-10" "-0021-08-11" "-0021-08-12" "-0021-08-13"
[226] "-0021-08-14" "-0021-08-15" "-0021-08-16" "-0021-08-17" "-0021-08-18"
[231] "-0021-08-19" "-0021-08-20" "-0021-08-21" "-0021-08-22" "-0021-08-23"
[236] "-0021-08-24" "-0021-08-25" "-0021-08-26" "-0021-08-27" "-0021-08-28"
[241] "-0021-08-29" "-0021-08-30" "-0021-08-31" "-0021-09-01" "-0021-09-02"
[246] "-0021-09-03" "-0021-09-04" "-0021-09-05" "-0021-09-06" "-0021-09-07"
[251] "-0021-09-08" "-0021-09-09" "-0021-09-10" "-0021-09-11" "-0021-09-12"
[256] "-0021-09-13" "-0021-09-14" "-0021-09-15" "-0021-09-16" "-0021-09-17"
[261] "-0021-09-18" "-0021-09-19" "-0021-09-20" "-0021-09-21" "-0021-09-22"
[266] "-0021-09-23" "-0021-09-24" "-0021-09-25" "-0021-09-26" "-0021-09-27"
[271] "-0021-09-28" "-0021-09-29" "-0021-09-30" "-0021-10-01" "-0021-10-02"
[276] "-0021-10-03" "-0021-10-04" "-0021-10-05" "-0021-10-06" "-0021-10-07"
[281] "-0021-10-08" "-0021-10-09" "-0021-10-10" "-0021-10-11" "-0021-10-12"
[286] "-0021-10-13" "-0021-10-14" "-0021-10-15" "-0021-10-16" "-0021-10-17"
[291] "-0021-10-18" "-0021-10-19" "-0021-10-20" "-0021-10-21" "-0021-10-22"
[296] "-0021-10-23" "-0021-10-24" "-0021-10-25" "-0021-10-26" "-0021-10-27"
[301] "-0021-10-28" "-0021-10-29" "-0021-10-30" "-0021-10-31" "-0021-11-01"
[306] "-0021-11-02" "-0021-11-03" "-0021-11-04" "-0021-11-05" "-0021-11-06"
[311] "-0021-11-07" "-0021-11-08" "-0021-11-09" "-0021-11-10" "-0021-11-11"
[316] "-0021-11-12" "-0021-11-13" "-0021-11-14" "-0021-11-15" "-0021-11-16"
[321] "-0021-11-17" "-0021-11-18" "-0021-11-19" "-0021-11-20" "-0021-11-21"
[326] "-0021-11-22" "-0021-11-23" "-0021-11-24" "-0021-11-25" "-0021-11-26"
[331] "-0021-11-27" "-0021-11-28" "-0021-11-29" "-0021-11-30" "-0021-12-01"
[336] "-0021-12-02" "-0021-12-03" "-0021-12-04" "-0021-12-05" "-0021-12-06"
[341] "-0021-12-07" "-0021-12-08" "-0021-12-09" "-0021-12-10" "-0021-12-11"
[346] "-0021-12-12" "-0021-12-13" "-0021-12-14" "-0021-12-15" "-0021-12-16"
[351] "-0021-12-17" "-0021-12-18" "-0021-12-19" "-0021-12-20" "-0021-12-21"
[356] "-0021-12-22" "-0021-12-23" "-0021-12-24" "-0021-12-25" "-0021-12-26"
[361] "-0021-12-27" "-0021-12-28" "-0021-12-29" "-0021-12-30" "-0021-12-31"

```

2.4. Contract

The `contract()` function operates as the opposite of `expand()`. It contracts a list of dates into the abbreviated annotation of **messydates**.

```
R> tibble::tibble(contract = contract(expand(dates_expand)))
```



```
# A tibble: 7 x 1
  contract
  <mdate>
1 2001-01-01 ~
2 2001-01 ~
3 2001 ~
4 2001-01-01..2001-02-02 ~
5 {2001-01-01,2001-02-02} ~
6 {2001-01-31,2001-02-28,2001-03-31,2001-04-30,2001-05-31,2001-06-30,2001-07-31~
7 -0021 ~
```

2.5. Coerce from messydates

Coercion functions coerce objects of `mdate` class to common date classes such as `Date`, `POSIXct`, and `POSIXlt`. Since `mdate` objects can hold multiple individual dates, an additional function must be passed as an argument so that multiple dates are “resolved” into a single date.

For example, one might wish to use the earliest possible date in any ranges of dates (`min`), the latest possible date (`max`), some notion of a central tendency (`mean`, `median`, or `modal`), or even a `random` selection from amongst the candidate dates.

These functions are particularly useful for use with existing methods and models, especially for checking the robustness of results.

```
R> tibble::tibble(min = as.Date(dates_expand, min),
+                 max = as.Date(dates_expand, max),
+                 median = as.Date(dates_expand, median),
+                 mean = as.Date(dates_expand, mean),
+                 modal = as.Date(dates_expand, modal),
+                 random = as.Date(dates_expand, random))

# A tibble: 7 x 6
  min      max      median      mean      modal      random
  <date>   <date>   <date>   <date>   <date>   <date>
1 2001-01-01 2001-01-01 2001-01-01 2001-01-01 2001-01-01 2001-01-01
2 2001-01-01 2001-01-31 2001-01-16 2001-01-16 2001-01-01 2001-01-26
3 2001-01-01 2001-12-31 2001-07-02 2001-07-02 2001-01-01 2001-11-13
4 2001-01-01 2001-02-02 2001-01-17 2001-01-17 2001-01-01 2001-01-12
5 2001-01-01 2001-02-02 2001-02-02 2001-01-17 2001-01-01 2001-01-01
6 2001-01-31 2001-12-31 2001-07-31 2001-07-15 2001-01-31 2001-12-31
7 -021-01-01 -021-12-31 -021-07-02 -021-07-02 -021-01-01 -021-06-15
```

2.6. Additional functionality

Several other functions are also offered in the **messydates** package.

For example, one can check various logical tests for messy date objects. `is_messydate()` tests whether the object inherits the `mdate` class. `is_intersecting()` tests whether there is any intersection between two messy dates. `is_element()` similarly tests whether a messy date can be found within a messy date range or set. `is_similar()` tests whether two dates contain similar components. `is_precise()` tests for whether date is precise.

```
R> is_messydate(as_messydate("2001-01-01"))
```

```
[1] TRUE
```

```
R> is_messydate(as.Date("2001-01-01"))
```

```
[1] FALSE
```

```
R> is_intersecting(as_messydate("2001-01"), as_messydate("2001-01-01..2001-02-22"))
```

Please specify 'approx_range' argument if you want approximate dates to also be expanded
Please specify 'approx_range' argument if you want approximate dates to also be expanded

```
[1] TRUE
```

```
R> is_intersecting(as_messydate("2001-01"), as_messydate("2001-02-01..2001-02-22"))
```

Please specify 'approx_range' argument if you want approximate dates to also be expanded
Please specify 'approx_range' argument if you want approximate dates to also be expanded

```
[1] FALSE
```

```
R> is_element(as_messydate("2001-01-01"), as_messydate("2001-01"))
```

Please specify 'approx_range' argument if you want approximate dates to also be expanded

```
[1] TRUE
```

```
R> is_element(as_messydate("2001-01-01"), as_messydate("2001-02"))
```

Please specify 'approx_range' argument if you want approximate dates to also be expanded

```
[1] FALSE
```

```
R> is_similar(as_messydate("2001-06-02"), as_messydate("2001-02-06"))
```

```
[1] TRUE
```

```
R> is_similar(as_messydate("2001-06-22"), as_messydate("2001-02-06"))
```

```
[1] FALSE
```

```
R> is_precise(as_messydate("2001-06-02"))
```

```
[1] TRUE
```

```
R> is_precise(as_messydate("2001-02"))
```

```
[1] FALSE
```

Additionally, one can perform intersection (`md_intersect()`) and union (`md_union()`) on, inter alia, messy date class objects. Or ‘join’ that retains all elements, even if duplicated, with `md_multiset`.

```
R> md_intersect(as_messydate("2001-01-01..2001-01-20"), as_messydate("2001-01"))
```

Please specify ‘approx_range’ argument if you want approximate dates to also be expanded
Please specify ‘approx_range’ argument if you want approximate dates to also be expanded

```
[1] "2001-01-01" "2001-01-02" "2001-01-03" "2001-01-04" "2001-01-05"
[6] "2001-01-06" "2001-01-07" "2001-01-08" "2001-01-09" "2001-01-10"
[11] "2001-01-11" "2001-01-12" "2001-01-13" "2001-01-14" "2001-01-15"
[16] "2001-01-16" "2001-01-17" "2001-01-18" "2001-01-19" "2001-01-20"
```

```
R> md_union(as_messydate("2001-01-01..2001-01-20"), as_messydate("2001-01"))
```

Please specify ‘approx_range’ argument if you want approximate dates to also be expanded
Please specify ‘approx_range’ argument if you want approximate dates to also be expanded

```
[1] "2001-01-01" "2001-01-02" "2001-01-03" "2001-01-04" "2001-01-05"
[6] "2001-01-06" "2001-01-07" "2001-01-08" "2001-01-09" "2001-01-10"
[11] "2001-01-11" "2001-01-12" "2001-01-13" "2001-01-14" "2001-01-15"
[16] "2001-01-16" "2001-01-17" "2001-01-18" "2001-01-19" "2001-01-20"
[21] "2001-01-21" "2001-01-22" "2001-01-23" "2001-01-24" "2001-01-25"
[26] "2001-01-26" "2001-01-27" "2001-01-28" "2001-01-29" "2001-01-30"
[31] "2001-01-31"
```

```
R> md_multiset(as_messydate("2001-01-01..2001-01-20"), as_messydate("2001-01"))
```

Please specify ‘approx_range’ argument if you want approximate dates to also be expanded
Please specify ‘approx_range’ argument if you want approximate dates to also be expanded

```
[1] "2001-01-01" "2001-01-02" "2001-01-03" "2001-01-04" "2001-01-05"
[6] "2001-01-06" "2001-01-07" "2001-01-08" "2001-01-09" "2001-01-10"
[11] "2001-01-11" "2001-01-12" "2001-01-13" "2001-01-14" "2001-01-15"
[16] "2001-01-16" "2001-01-17" "2001-01-18" "2001-01-19" "2001-01-20"
```

```
[21] "2001-01-01" "2001-01-02" "2001-01-03" "2001-01-04" "2001-01-05"
[26] "2001-01-06" "2001-01-07" "2001-01-08" "2001-01-09" "2001-01-10"
[31] "2001-01-11" "2001-01-12" "2001-01-13" "2001-01-14" "2001-01-15"
[36] "2001-01-16" "2001-01-17" "2001-01-18" "2001-01-19" "2001-01-20"
[41] "2001-01-21" "2001-01-22" "2001-01-23" "2001-01-24" "2001-01-25"
[46] "2001-01-26" "2001-01-27" "2001-01-28" "2001-01-29" "2001-01-30"
[51] "2001-01-31"
```

As well, some arithmetic operations are available for messydates. For instance, one can add or subtract one year to all messy dates in a vector.

```
R> tibble::tibble(date = dates_expand, add = dates_expand + "1 day", subtract = dates_exp
```

```
# A tibble: 7 x 3
```

	date <mdate>	add <mdate>	subtract <mdate>
1	2001-01-01	2001-01-02	~ 2000-01~
2	2001-01?	2001-01-02..2001-02-01	~ 2000-01~
3	2001	2001-01-02..2002-01-01	~ 2000-01~
4	2001-01-01..2001-02-02	2001-01-02..2001-02-03	~ 2000-01~
5	{2001-01-01,2001-02-02}	{2001-01-02,2001-02-03}	~ {2000-0~
6	2001-XX-31	{2001-02-01,2001-03-01,2001-04-01,2001-05-01}	~ {2000-0~
7	-0021	-0020-12-31..-0021-12-30	~ -0022 ~

2.7. Case Study - 2001 Battles

Dates, even for some recent events, can be messy. Take, for example, the dates of battles in 2001 according to [Wikipedia](#). The dates of these battles are often uncertain, with different levels of date precision being reported. **messydates** facilitates working with these dates.

```
R> battles <- tibble::tribble(~Battle, ~Date, ~Parties,
+                               "Operation MH-2", "2001 March 8", "MK-National Liberation Army",
+                               "2001 Bangladesh-India border clashes", "2001-04-16..2001-04-17", "India",
+                               "Operation Vaksince", "25-5-2001", "MK-National Liberation Army",
+                               "Alkhan-Kala operation", "2001-6-22..2001-6-28", "RU-Chechen",
+                               "Battle of Vedenov", "2001-8-13..2001-8-26", "RU-Chechen Insurgents",
+                               "Operation Crescent Wind", "2001-10-7..2001-12-7", "US/UK-Taliban",
+                               "Operation Rhino", "2001-10-19..2001-10-20", "US-Taliban",
+                               "Battle of Mazar-e-Sharif", "2001-11-9", "US/Northern Alliance",
+                               "Siege of Kunduz", "2001-11-11..2001-11-23", "US/Northern Alliance",
+                               "Battle of Herat", "Twelve of November of two thousand and two", "US/Northern Alliance",
+                               "Battle of Kabul", "2001-11-13..2001-11-14", "US/Northern Alliance",
+                               "Battle of Tarin Kowt", "2001-11-13:2001-11-14", "US/Eastern Front",
+                               "Operation Trent", "2001-11-15..2001-11-30", "US/UK-Taliban",
+                               "Battle of Kandahar", "2001-11-22..2001-12-07", "US/AU/Eastern Front",
+                               "Battle of Qala-i-Jangi", "2001-11-25:2001-12-01", "US/UK/Northern Alliance")
```

```

+           "Battle of Tora Bora", "2001-12-12..2001-12-17", "US/Northern
+           "Battle of Shawali Kowt", "2001-12-3", "US/Eastern Alliance-T
+           "Battle of Sayyd Alma Kalay", "2001-12-4", "US/Eastern Allian
+           "Battle of Amami-Oshima", "2001-12-22", "JP-KP",
+           "Tsotsin-Yurt operation", "2001-12-30:2002-01-03", "RU-Cheche
R> battles$Date = as_messydate(battles$Date)
R> tibble::tibble(battles)

```

```
# A tibble: 20 x 3
```

Battle <chr>	Date <mdate>	Parties <chr>
1 Operation MH-2	2001-03-08	MK-National Li~
2 2001 Bangladesh-India border clashes	2001-04-16..2001-04-20	BD-ID
3 Operation Vaksince	2001-05-25	MK-National Li~
4 Alkhan-Kala operation	2001-06-22..2001-06-28	RU-Chechen Rep~
5 Battle of Vedenov	2001-08-13..2001-08-26	RU-Chechen Ins~
6 Operation Crescent Wind	2001-10-7..2001-12?	US/UK-Taliban
7 Operation Rhino	2001-10-19..2001-10-20	US-Taliban
8 Battle of Mazar-e-Sharif	2001-11-09	US/Northern Al~
9 Siege of Kunduz	2001-11-11..2001-11-23	US/Northern Al~
10 Battle of Herat	2021-11-12	US/Northern Al~
11 Battle of Kabul	2001-11-13..2001-11-14	US/Northern Al~
12 Battle of Tarin Kowt	2001-11-13..2001-11-14	US/Eastern All~
13 Operation Trent	2001-11-~15..2001-11-~30	US/UK-Taliban/~
14 Battle of Kandahar	2001-11-22..2001-12-07	US/AU/Eastern ~
15 Battle of Qala-i-Jangi	2001-11-25..2001-12-01	US/UK/Northern~
16 Battle of Tora Bora	2001-12-12..2001-12-17	US/Northern Al~
17 Battle of Shawali Kowt	2001-12-03	US/Eastern All~
18 Battle of Sayyd Alma Kalay	2001-12-04	US/Eastern All~
19 Battle of Amami-Oshima	2001-12-22	JP-KP
20 Tsotsin-Yurt operation	2001-12-30..2002-01-03	RU-Chechen Ins~

Getting the timing can be important for researchers, however, when faced with date imprecision, researchers usually have to choose between making arbitrary choices (e.g. adding “-01-01” to all incomplete dates) or work imprecise dates (i.e. year only). Yet, both choices may lead to bias results. This is especially true if researchers’ are looking to generate inferences where getting “timing” right is important.

Assume, for example, certain researcher is interested in the relationship between the United States (US) being a party in a conflict and the duration of the conflict in 2001. The researcher might theorize that conflicts involving the US have a shorter duration since the US has the most powerful military in the world. As well, this relationship could be mediated by the number of parties involved in the conflict. That is, the number of actors involved in a conflict could have an effect on conflict time. Using **messydates**, we can create two different date variables in the battles data to represent conflict time, one with an arbitrary cut of point (difference between the minimal and maximal values) and the other with random values (difference between two random values in the range for uncertain or approximate dates). These will be our dependent variables. We can also create a dummy variable for whether the

US was, or not, involved in the conflict to be our main independent variable. As a control, we also code the number of actors in the conflict. With these variables in hand, we build two linear regression models correlating each of the dependent variables to conflict time.

```
R> set.seed(3333)
R> battles <- battles %>%
+   mutate(arbitrary = as.numeric(as.Date(Date, max) - as.Date(Date, min)),
+         random = ifelse(grepl("\\?|\\~", Date),
+                         abs(as.Date(Date, random) - as.Date(Date, random)),
+                         arbitrary),
+         US_party = ifelse(grepl("US", Parties), 1, 0),
+         n_actors = c(2, 2, 2, 2, 2, 3, 2, 4, 4, 4, 3, 3, 4, 4, 5, 4, 3, 3, 2, 2))
R> arbitrary <- lm(arbitrary ~ US_party + n_actors, battles)
R> random <- lm(random ~ US_party + n_actors, battles)
R> library(stargazer)
R> stargazer::stargazer(arbitrary, random, type = "text")
```

	Dependent variable:	
	arbitrary (1)	random (2)
US_party	10.802 (14.399)	1.419 (6.484)
n_actors	-2.479 (7.239)	0.670 (3.260)
Constant	8.815 (16.241)	2.517 (7.314)
Observations	20	20
R2	0.040	0.023
Adjusted R2	-0.073	-0.092
Residual Std. Error (df = 17)	19.467	8.766
F Statistic (df = 2; 17)	0.352	0.199
Note:	*p<0.1; **p<0.05; ***p<0.01	

Notice how the regression coefficients change when we pick random values within the range for the uncertain and approximate dates in the battles data, in comparison to setting arbitrary cut off points. Although not statistically significant, the coefficient for US being a party in a conflict goes from being positive when we use arbitrary cut off points for uncertain dates, to negative when we use random values within that range. In this case, it is hard to say

whether the relationship between the US being a part of a battle in 2001 and the time of the conflict is positive or not.

3. Acknowledgements

We would like to thank the Swiss National Science Foundation. This work was supported by grant number 188976.

References

- Eddelbuettel D (2019). “Anytime: Easier Date and Time Conversion.” p. 4.
- Grolemund G, Wickham H (2011). “Dates and Times Made Easy with Lubridate.” **40**(3), 1–25. ISSN 1548-7660. doi:10.18637/jss.v040.i03. URL <http://www.jstatsoft.org/v40/i03/>.

Affiliation:

James Hollway
Graduate Institute of
International and Development Studies
Chemin Eugène-Rigot 2A
PO Box 1672
1211 Geneva 1
Switzerland
E-mail: james.hollway@graduateinstitute.ch
URL: <http://jameshollway.com>