Example 1

Your name

1/22/2020

This example shows how to convert a Column with string dates and empty spaces to three columns with month, day, and year. Using `lubridate` to convert it failed, so we chose to try POSIXct from R base.

```
#
# data
d1 = read.csv("StudyArea_SmallFile.csv")
str(d1)
```

```
## 'data.frame':    7154 obs. of  21 variables:
##  $ FID       : int  1603 1605 1608 1647 1668 1673 1675 1677 1680 1682 ...
##  $ ORGANIZATI: Factor w/ 6 levels "BIA","BLM","BOR",..: 5 5 5 5 5 5 5 5 5 5 ...
##  $ FIRENAME  : Factor w/ 5880 levels " "," GRIZZLY",..: 435 3502 3676 5865 2239 2911 1136 1137 1136 1
##  $ FIRENUMBER: Factor w/ 697 levels " ","1","10","100",..: 254 255 256 297 316 326 329 331 332 338 .
##  $ FIRECODE  : Factor w/ 5162 levels " ","1009","1016",..: 128 129 130 161 165 168 170 171 172 175 .
##  $ CAUSE     : Factor w/ 3 levels "Human","Natural",..: 1 1 1 1 1 1 1 1 2 ...
##  $ SPECCAUSE : int  30 30 30 30 10 10 30 10 0 30 ...
##  $ STATCAUSE : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ SIZECLASS : Factor w/ 6 levels "A","B","C","D",..: 5 6 5 5 6 5 6 6 6 5 ...
##  $ FIRETYPE  : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ YEAR_     : int  1988 1986 1986 2002 2000 2000 2002 2000 1991 1991 ...
##  $ STARTDATED: Factor w/ 2858 levels "1/1/94 0:00",..: 238 396 885 205 354 382 384 724 406 1503 ...
##  $ CONTRDATED: Factor w/ 3352 levels "","1/1/94 0:00",..: 636 793 1195 626 658 772 757 1249 782 1848
##  $ OUTDATED  : Factor w/ 3527 levels "","1/1/02 0:00",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ STATE     : Factor w/ 11 levels "Arizona","California",..: 1 1 5 1 1 1 1 1 1 1 ...
##  $ STATE_FIPS: int  4 4 30 4 4 4 4 4 4 4 ...
##  $ DLATITUDE : num  31.6 32.5 47.5 31.7 31.5 ...
##  $ DLONGITUDE: num  -112 -112 -111 -111 -112 ...
##  $ TOTALACRES: num  1500 10390 1400 1035 5700 ...
##  $ TRPGENCAUS: int  0 0 0 0 0 0 0 0 0 0 ...
##  $ TRPSPECCAU: int  0 0 0 0 0 0 0 0 0 0 ...
```

```
# subset cols
d1 = subset(d1,select=c(FIRENAME,YEAR_,OUTDATED,STATE,TOTALACRES))
dim(d1)
```

```
## [1] 7154    5
```

```
head(d1)
```

```
##     FIRENAME YEAR_ OUTDATED   STATE TOTALACRES
## 1 BIG BERTHA  1988          Arizona       1500
## 2     MORMON  1986          Arizona      10390
## 3      NORTH  1986          Montana       1400
## 4     YELLOW  2002          Arizona       1035
## 5        GUS  2000          Arizona       5700
## 6       LANE  2000          Arizona       2750
```

```
#
tail(d1)
```

```
##             FIRENAME YEAR_      OUTDATED      STATE TOTALACRES
```

```
## 7149 Lone Mountain 1  2014 10/31/14 0:00 Washington       2770
## 7150       Wolverine  2015 11/30/15 0:00 Washington       2600
## 7151         Goodell  2015  12/8/15 0:00 Washington       7107
## 7152          BEAVER  1985 11/15/85 0:00 Washington       1170
## 7153        Paradise  2015 11/18/15 0:00 Washington       2815
## 7154      Hayes Fire  2016  10/9/16 0:00 Washington       2475
```

```
n = nrow(d1)
n
```

```
## [1] 7154
```

```
# split date, time.
# install.packages("stringr")
library(stringr)
a1 = str_split(d1$OUTDATED," ")
head(a1)
```

```
## [[1]]
## [1] ""
##
## [[2]]
## [1] ""
##
## [[3]]
## [1] ""
##
## [[4]]
## [1] ""
##
## [[5]]
## [1] ""
##
## [[6]]
## [1] ""
```

```
tail(a1)
```

```
## [[1]]
## [1] "10/31/14" "0:00"
##
## [[2]]
## [1] "11/30/15" "0:00"
##
## [[3]]
## [1] "12/8/15" "0:00"
##
## [[4]]
## [1] "11/15/85" "0:00"
##
## [[5]]
## [1] "11/18/15" "0:00"
##
## [[6]]
## [1] "10/9/16" "0:00"
```

```r
class(a1)
```

```
## [1] "list"
```

```r
# a1 has empty spaces or two string
length(a1[[1]])
```

```
## [1] 1
```

```r
length(a1[[n]])
```

```
## [1] 2
```

```r
# first and only element in a1 is empty
# last element in a1 is a set of two strings (one date, one time)

# vectors filled with zeros
year = rep(0,n)
month = rep(0,n)
day = rep(0,n)

# use Rbase POSIXct() to convert string to POSIXct object, it gives NA if a1 is empty
# use lubridate functions year, month, day to extract date components as numeric
#
library(lubridate)
```

```
##
## Attaching package: 'lubridate'
```

```
## The following object is masked from 'package:base':
##
##     date
```

```r
for(i in 1:n)
{
  b1 = a1[[i]][1]
  c1 = as.POSIXct(b1,format = "%m/%d/%y")
  c1 = as.Date(c1)
  year[i] = year(c1)
  month[i] = month(c1)
  day[i] = day(c1)
}

# add columns to dataframe d1
head(d1)
```

```
##     FIRENAME YEAR_ OUTDATED   STATE TOTALACRES
## 1 BIG BERTHA  1988          Arizona       1500
## 2     MORMON  1986          Arizona      10390
## 3      NORTH  1986          Montana       1400
## 4     YELLOW  2002          Arizona       1035
## 5        GUS  2000          Arizona       5700
## 6       LANE  2000          Arizona       2750
```

```r
d1$Year = year
d1$Month = month
d1$Day = day
#
```

```r
tail(d1)
```

```
##              FIRENAME YEAR_      OUTDATED      STATE TOTALACRES Year Month Day
## 7149 Lone Mountain 1  2014 10/31/14 0:00 Washington       2770 2014    10  31
## 7150       Wolverine  2015 11/30/15 0:00 Washington       2600 2015    11  30
## 7151         Goodell  2015  12/8/15 0:00 Washington       7107 2015    12   8
## 7152          BEAVER  1985 11/15/85 0:00 Washington       1170 1985    11  15
## 7153        Paradise  2015 11/18/15 0:00 Washington       2815 2015    11  18
## 7154      Hayes Fire  2016  10/9/16 0:00 Washington       2475 2016    10   9
```

```r
head(d1)
```

```
##     FIRENAME YEAR_ OUTDATED   STATE TOTALACRES Year Month Day
## 1 BIG BERTHA  1988          Arizona       1500   NA    NA  NA
## 2     MORMON  1986          Arizona      10390   NA    NA  NA
## 3      NORTH  1986          Montana       1400   NA    NA  NA
## 4     YELLOW  2002          Arizona       1035   NA    NA  NA
## 5        GUS  2000          Arizona       5700   NA    NA  NA
## 6       LANE  2000          Arizona       2750   NA    NA  NA
```

The resulting dataframe shows the new columns Year, Month, and Day, created from OUTDATED. When there is not string in OUTDATED these new columns show NAs.