HW4

Ben Gaudiosi

February 21, 2018

10.5

5

It converts tibbles to a dataframe. You may want to use this with legacy code or if you want to flatten data into a dataframe for any other purpose.

12.6.1

```
suppressMessages(library(tidyverse))

who1 <- who %>%
  gather(new_sp_m014:newrel_f65, key = "key", value = "cases", na.rm = TRUE)
glimpse(who1)
```

```
## Observations: 76,046
## Variables: 6
## $ country <chr> "Afghanistan", "Afghanistan", "Afghanistan", "Afghanis...
## $ iso2    <chr> "AF", "AF", "AF", "AF", "AF", "AF", "AF", "AF", "AF", ...
## $ iso3    <chr> "AFG", "AFG", "AFG", "AFG", "AFG", "AFG", "AFG", "AFG"...
## $ year    <int> 1997, 1998, 1999, 2000, 2001, 2002, 2003, 2004, 2005, ...
## $ key     <chr> "new_sp_m014", "new_sp_m014", "new_sp_m014", "new_sp_m...
## $ cases   <int> 0, 30, 8, 52, 129, 90, 127, 139, 151, 193, 186, 187, 2...
```

```
who2 <- who1 %>%
 mutate(key = stringr::str_replace(key, "newrel", "new_rel"))

who3 <- who2 %>%
  separate(key, c("new", "type", "sexage"), sep = "_")
who3
```

```
## # A tibble: 76,046 x 8
##    country     iso2  iso3   year new   type  sexage cases
##    <chr>       <chr> <chr> <int> <chr> <chr> <chr>  <int>
##  1 Afghanistan AF    AFG    1997 new   sp    m014       0
##  2 Afghanistan AF    AFG    1998 new   sp    m014      30
##  3 Afghanistan AF    AFG    1999 new   sp    m014       8
##  4 Afghanistan AF    AFG    2000 new   sp    m014      52
##  5 Afghanistan AF    AFG    2001 new   sp    m014     129
##  6 Afghanistan AF    AFG    2002 new   sp    m014      90
##  7 Afghanistan AF    AFG    2003 new   sp    m014     127
##  8 Afghanistan AF    AFG    2004 new   sp    m014     139
##  9 Afghanistan AF    AFG    2005 new   sp    m014     151
## 10 Afghanistan AF    AFG    2006 new   sp    m014     193
## # ... with 76,036 more rows
```

```r
who3 %>%
  count(new)
```

```
## # A tibble: 1 x 2
##   new       n
##   <chr> <int>
## 1 new   76046
```

```r
who4 <- who3 %>%
  select(-new, -iso2, -iso3)

who5 <- who4 %>%
  separate(sexage, c("sex", "age"), sep = 1)
who5
```

```
## # A tibble: 76,046 x 6
##    country      year type  sex   age   cases
##    <chr>       <int> <chr> <chr> <chr> <int>
##  1 Afghanistan  1997 sp    m     014       0
##  2 Afghanistan  1998 sp    m     014      30
##  3 Afghanistan  1999 sp    m     014       8
##  4 Afghanistan  2000 sp    m     014      52
##  5 Afghanistan  2001 sp    m     014     129
##  6 Afghanistan  2002 sp    m     014      90
##  7 Afghanistan  2003 sp    m     014     127
##  8 Afghanistan  2004 sp    m     014     139
##  9 Afghanistan  2005 sp    m     014     151
## 10 Afghanistan  2006 sp    m     014     193
## # ... with 76,036 more rows
```

3

```r
select(who3, country, iso2, iso3) %>%
  distinct() %>%
  group_by(country) %>%
  filter(n() > 1)
```

```
## # A tibble: 0 x 3
## # Groups:   country [0]
## # ... with 3 variables: country <chr>, iso2 <chr>, iso3 <chr>
```

4

```r
who5 %>%
  group_by(country, year, sex) %>%
  summarize(cases = sum(cases)) %>%
  unite(country_sex, country, sex, remove = FALSE) %>%
  ggplot(aes(x = year, y = cases, group = country_sex, colour = sex)) +
  geom_line()
```

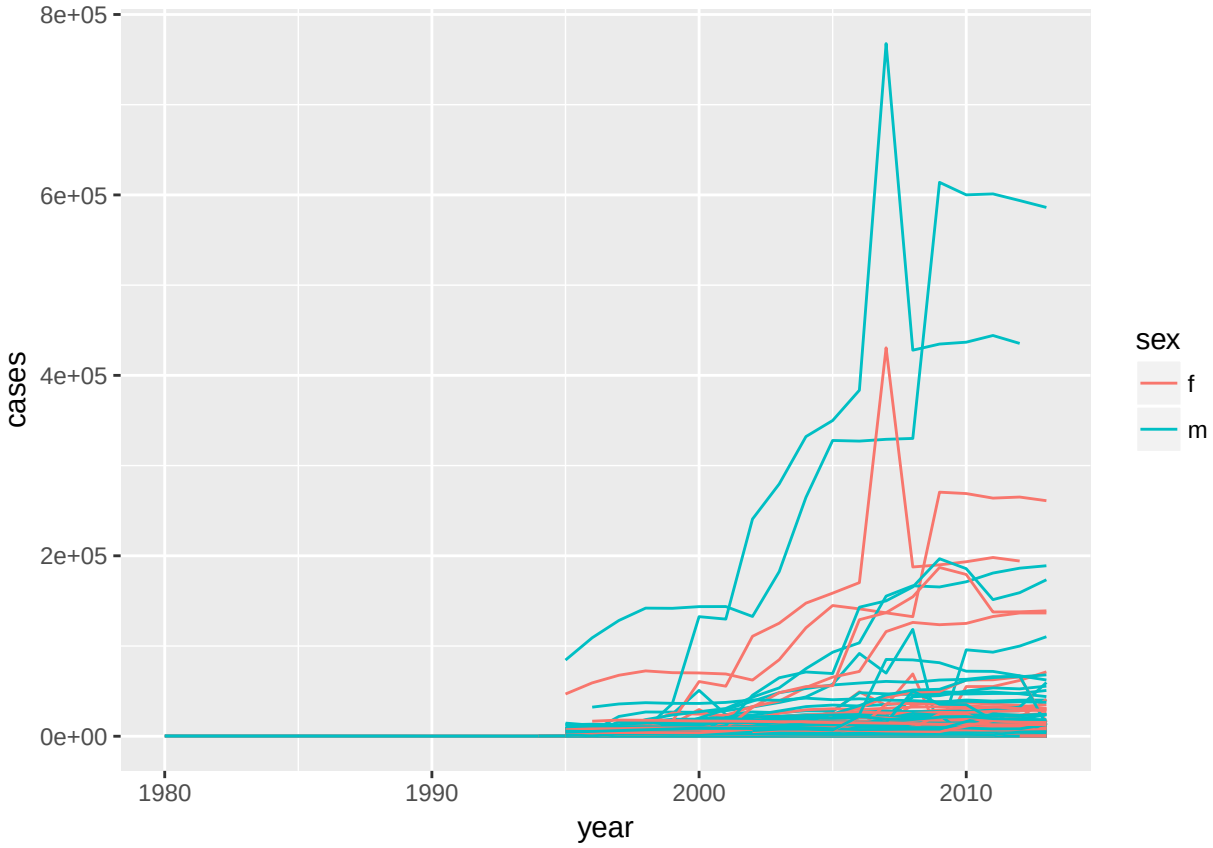

table 4 -> table 6

```
library(foreign)
library(stringr)
library(plyr)
```

```
## ------------------------------------------------------------------------
## You have loaded plyr after dplyr - this is likely to cause problems.
## If you need functions from both plyr and dplyr, please load plyr first, then dplyr:
## library(plyr); library(dplyr)
## ------------------------------------------------------------------------
##
## Attaching package: 'plyr'
## The following objects are masked from 'package:dplyr':
##
##     arrange, count, desc, failwith, id, mutate, rename, summarise,
##     summarize
## The following object is masked from 'package:purrr':
##
##     compact
```

```
library(reshape2)
```

```
##
## Attaching package: 'reshape2'

## The following object is masked from 'package:tidyr':
##
##     smiths
```

```
pew <- read.spss("pew.sav")
```

```
## re-encoding from CP1252

## Warning in read.spss("pew.sav"): Undeclared level(s) 2, 3, 4, 9 added in
## variable: density3

## Warning in read.spss("pew.sav"): Duplicated levels in factor denom:
## Electronic ministries

## Warning in read.spss("pew.sav"): Undeclared level(s) 1, 2, 3, 4, 5, 6, 7,
## 8, 9, 10, 11, 12, 14, 16, 23, 33 added in variable: children

## Warning in read.spss("pew.sav"): Undeclared level(s) 18, 19, 20, 21, 22,
## 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41,
## 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60,
## 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79,
## 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96 added in
## variable: age
```

```
pew <- as.data.frame(pew)


religion <- pew[c("q16", "reltrad", "income")]
religion$reltrad <- as.character(religion$reltrad)
religion$reltrad <- str_replace(religion$reltrad, " Churches", "")
religion$reltrad <- str_replace(religion$reltrad, " Protestant", " Prot")
religion$reltrad[religion$q16 == " Atheist (do not believe in God) "] <- "Atheist"
religion$reltrad[religion$q16 == " Agnostic (not sure if there is a God) "] <- "Agnostic"
religion$reltrad <- str_trim(religion$reltrad)
religion$reltrad <- str_replace_all(religion$reltrad, " \\(.*?\\)", "")

religion$income <- c("Less than $10,000" = "<$10k",
  "10 to under $20,000" = "$10-20k",
  "20 to under $30,000" = "$20-30k",
  "30 to under $40,000" = "$30-40k",
  "40 to under $50,000" = "$40-50k",
  "50 to under $75,000" = "$50-75k",
  "75 to under $100,000" = "$75-100k",
  "100 to under $150,000" = "$100-150k",
  "$150,000 or more" = ">150k",
  "Don't know/Refused (VOL)" = "Don't know/refused")[religion$income]

religion$income <- factor(religion$income, levels = c("<$10k", "$10-20k", "$20-30k", "$30-40k", "$40-50
  "$75-100k", "$100-150k", ">150k", "Don't know/refused"))

counts <- count(religion, c("reltrad", "income"))
names(counts)[1] <- "religion"

# Convert into the form in which I originally saw it -------------------------
```

```
raw <- dcast(counts, religion ~ income)
```

```
## Using freq as value column: use value.var to override.
```

```
unordered <- raw %>% gather(key='Income',value='Frequency', 2:11)
fixed <- unordered %>% arrange(religion)
raw
```

```
##                   religion <$10k $10-20k $20-30k $30-40k $40-50k $50-75k
## 1                 Agnostic    27      34      60      81      76     137
## 2                  Atheist    12      27      37      52      35      70
## 3                 Buddhist    27      21      30      34      33      58
## 4                 Catholic   418     617     732     670     638    1116
## 5       Don't know/refused    15      14      15      11      10      35
## 6         Evangelical Prot   575     869    1064     982     881    1486
## 7                    Hindu     1       9       7       9      11      34
## 8  Historically Black Prot   228     244     236     238     197     223
## 9        Jehovah's Witness    20      27      24      24      21      30
## 10                  Jewish    19      19      25      25      30      95
## 11           Mainline Prot   289     495     619     655     651    1107
## 12                  Mormon    29      40      48      51      56     112
## 13                  Muslim     6       7       9      10       9      23
## 14                Orthodox    13      17      23      32      32      47
## 15         Other Christian     9       7      11      13      13      14
## 16            Other Faiths    20      33      40      46      49      63
## 17   Other World Religions     5       2       3       4       2       7
## 18            Unaffiliated   217     299     374     365     341     528
##    $75-100k $100-150k >150k Don't know/refused
## 1       122       109    84                 96
## 2        73        59    74                 76
## 3        62        39    53                 54
## 4       949       792   633               1489
## 5        21        17    18                116
## 6       949       723   414               1529
## 7        47        48    54                 37
## 8       131        81    78                339
## 9        15        11     6                 37
## 10       69        87   151                162
## 11      939       753   634               1328
## 12       85        49    42                 69
## 13       16         8     6                 22
## 14       38        42    46                 73
## 15       18        14    12                 18
## 16       46        40    41                 71
## 17        3         4     4                  8
## 18      407       321   258                597
```

```
head(fixed, 20)
```

```
##    religion            Income Frequency
## 1  Agnostic             <$10k        27
## 2  Agnostic            $10-20k        34
## 3  Agnostic            $20-30k        60
## 4  Agnostic            $30-40k        81
## 5  Agnostic            $40-50k        76
```

```
## 6  Agnostic            $50-75k       137
## 7  Agnostic           $75-100k       122
## 8  Agnostic          $100-150k       109
## 9  Agnostic              >150k        84
## 10 Agnostic Don't know/refused        96
## 11  Atheist              <$10k        12
## 12  Atheist            $10-20k        27
## 13  Atheist            $20-30k        37
## 14  Atheist            $30-40k        52
## 15  Atheist            $40-50k        35
## 16  Atheist            $50-75k        70
## 17  Atheist           $75-100k        73
## 18  Atheist          $100-150k        59
## 19  Atheist              >150k        74
## 20  Atheist Don't know/refused        76
```

table 7 -> table 8

```
bb <- read_csv("billboard.csv")
```

```
## Parsed with column specification:
## cols(
##   .default = col_integer(),
##   artist.inverted = col_character(),
##   track = col_character(),
##   time = col_time(format = ""),
##   genre = col_character(),
##   date.entered = col_date(format = ""),
##   date.peaked = col_date(format = ""),
##   x66th.week = col_character(),
##   x67th.week = col_character(),
##   x68th.week = col_character(),
##   x69th.week = col_character(),
##   x70th.week = col_character(),
##   x71st.week = col_character(),
##   x72nd.week = col_character(),
##   x73rd.week = col_character(),
##   x74th.week = col_character(),
##   x75th.week = col_character(),
##   x76th.week = col_character()
## )
```

```
## See spec(...) for full column specifications.
```

```
bb.1 <- bb %>% gather(key="week", value = "rank", -year, -artist.inverted, -track, -time, -genre, -date
bb.2 <- bb.1 %>% select(year, artist=artist.inverted, time, track, date = date.entered, week, rank )
bb.3 <- bb.2 %>% arrange(track)
bb.4 <- bb.3 %>% filter(!is.na(rank))
bb.5 <- bb.4 %>% separate(week, into=c("A", "B", "C"), sep=c(1, -7), convert=TRUE)
bb.6 <- bb.5 %>% select(-A, -C)
bb.7 <- bb.6 %>%   dplyr::rename(week = B)
bb.8 <- bb.7 %>% arrange(artist, track)
bb.9 <- bb.8 %>% mutate(date = date + (week-1)*7 )
bb.10 <- bb.9 %>% mutate(rank = as.integer(rank))
```

```r
head(bb.10, 20)
```

```
## # A tibble: 20 x 7
##     year artist       time   track                   date        week  rank
##    <int> <chr>        <time> <chr>                   <date>     <int> <int>
##  1  2000 2 Pac        04:22  Baby Don't Cry (Keep ~ 2000-02-26     1    87
##  2  2000 2 Pac        04:22  Baby Don't Cry (Keep ~ 2000-03-04     2    82
##  3  2000 2 Pac        04:22  Baby Don't Cry (Keep ~ 2000-03-11     3    72
##  4  2000 2 Pac        04:22  Baby Don't Cry (Keep ~ 2000-03-18     4    77
##  5  2000 2 Pac        04:22  Baby Don't Cry (Keep ~ 2000-03-25     5    87
##  6  2000 2 Pac        04:22  Baby Don't Cry (Keep ~ 2000-04-01     6    94
##  7  2000 2 Pac        04:22  Baby Don't Cry (Keep ~ 2000-04-08     7    99
##  8  2000 2Ge+her      03:15  The Hardest Part Of B~ 2000-09-02     1    91
##  9  2000 2Ge+her      03:15  The Hardest Part Of B~ 2000-09-09     2    87
## 10  2000 2Ge+her      03:15  The Hardest Part Of B~ 2000-09-16     3    92
## 11  2000 3 Doors Down 03:53  Kryptonite             2000-04-08     1    81
## 12  2000 3 Doors Down 03:53  Kryptonite             2000-04-15     2    70
## 13  2000 3 Doors Down 03:53  Kryptonite             2000-04-22     3    68
## 14  2000 3 Doors Down 03:53  Kryptonite             2000-04-29     4    67
## 15  2000 3 Doors Down 03:53  Kryptonite             2000-05-06     5    66
## 16  2000 3 Doors Down 03:53  Kryptonite             2000-05-13     6    57
## 17  2000 3 Doors Down 03:53  Kryptonite             2000-05-20     7    54
## 18  2000 3 Doors Down 03:53  Kryptonite             2000-05-27     8    53
## 19  2000 3 Doors Down 03:53  Kryptonite             2000-06-03     9    51
## 20  2000 3 Doors Down 03:53  Kryptonite             2000-06-10    10    51
```