Social Security Disability Case Study

Samrawit Feleke

12/26/2018

R Markdown

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see http://rmarkdown.rstudio.com.

When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

Coal dataset

Data Wrangling in R

```
#Read coal dataset
Social_Security <- read_csv("Social_Security_Disability.csv")

## Parsed with column specification:
## cols(
##   .default = col_integer(),
##   Fiscal_Year = col_character()
## )

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

#remove 1st column value
Social_Security <- Social_Security[,-1]
head(Social_Security)

## # A tibble: 6 x 25
##   Fiscal_Year Oct_Total Oct_Internet Nov_Total Nov_Internet Dec_Total
##   <chr>           <int>        <int>     <int>        <int>     <int>
## 1 FY08           176407        15082    204287        17301    151687
## 2 FY09           244781        32578    181161        25620    176107
## 3 FY10           286598        65533    213297        50098    198733
## 4 FY11           299033        92856    209553        63424    215239
## 5 FY12           227456        86811    200140        71175    254766
## 6 FY13           224624        92542    249910       107053    188183
## # ... with 19 more variables: Dec_Internet <int>, Jan_Total <int>,
## #   Jan_Internet <int>, Feb_Total <int>, Feb_Internet <int>,
## #   Mar_Total <int>, Mar_Internet <int>, Apr_Total <int>,
## #   Apr_Internet <int>, May_Total <int>, May_Internet <int>,
## #   June_Total <int>, June_Internet <int>, July_Total <int>,
```

```
## #   July_Internet <int>, August_Total <int>, August_Internet <int>,
## #   Sept_Internet <int>, Sept_Total <int>

# column names
colnames(Social_Security)

##  [1] "Fiscal_Year"     "Oct_Total"       "Oct_Internet"
##  [4] "Nov_Total"       "Nov_Internet"    "Dec_Total"
##  [7] "Dec_Internet"    "Jan_Total"       "Jan_Internet"
## [10] "Feb_Total"       "Feb_Internet"    "Mar_Total"
## [13] "Mar_Internet"    "Apr_Total"       "Apr_Internet"
## [16] "May_Total"       "May_Internet"    "June_Total"
## [19] "June_Internet"   "July_Total"      "July_Internet"
## [22] "August_Total"    "August_Internet" "Sept_Internet"
## [25] "Sept_Total"
```

Structure of the dataset

```
glimpse(Social_Security)

## Observations: 10
## Variables: 25
## $ Fiscal_Year     <chr> "FY08", "FY09", "FY10", "FY11", "FY12", "FY13"...
## $ Oct_Total       <int> 176407, 244781, 286598, 299033, 227456, 224624...
## $ Oct_Internet    <int> 15082, 32578, 65533, 92856, 86811, 92542, 9840...
## $ Nov_Total       <int> 204287, 181161, 213297, 209553, 200140, 249910...
## $ Nov_Internet    <int> 17301, 25620, 50098, 63424, 71175, 107053, 117...
## $ Dec_Total       <int> 151687, 176107, 198733, 215239, 254766, 188183...
## $ Dec_Internet    <int> 14321, 27174, 44512, 62877, 91424, 79719, 8337...
## $ Jan_Total       <int> 162966, 249062, 265665, 264286, 221146, 199588...
## $ Jan_Internet    <int> 18391, 57908, 68843, 84944, 85848, 93703, 1253...
## $ Feb_Total       <int> 228623, 221368, 225319, 223625, 228519, 219604...
## $ Feb_Internet    <int> 26034, 50408, 58465, 71314, 83576, 101878, 108...
## $ Mar_Total       <int> 190716, 235360, 243266, 246630, 299267, 285923...
## $ Mar_Internet    <int> 21064, 53592, 62198, 77916, 112104, 129415, 11...
## $ Apr_Total       <int> 194403, 234304, 298065, 300359, 233685, 224804...
## $ Apr_Internet    <int> 22372, 53675, 76573, 94722, 88330, 101619, 112...
## $ May_Total       <int> 226549, 281343, 239409, 241673, 239503, 269955...
## $ May_Internet    <int> 26337, 65822, 65780, 77603, 93826, 123440, 134...
## $ June_Total      <int> 193094, 237329, 231964, 233351, 284136, 223238...
## $ June_Internet   <int> 22551, 54285, 67163, 79925, 113613, 104146, 11...
## $ July_Total      <int> 181552, 285172, 300442, 292949, 221745, 204072...
## $ July_Internet   <int> 22728, 66565, 92957, 105276, 91323, 98326, 106...
## $ August_Total    <int> 245429, 240611, 248284, 237555, 298458, 281828...
## $ August_Internet <int> 30580, 54915, 75535, 86514, 119795, 135423, 13...
## $ Sept_Internet   <int> 24141, 52687, 73403, 103564, 93375, 104270, 10...
## $ Sept_Total      <int> 186750, 228692, 238965, 280913, 230648, 214004...
```

Transform to long dataset

```
Social_Security_long <- gather(Social_Security, month, applications, -
Fiscal_Year)
```

```
# View result
print(Social_Security_long, n=20)
```

```
## # A tibble: 240 x 3
##    Fiscal_Year month        applications
##    <chr>       <chr>               <int>
##  1 FY08        Oct_Total          176407
##  2 FY09        Oct_Total          244781
##  3 FY10        Oct_Total          286598
##  4 FY11        Oct_Total          299033
##  5 FY12        Oct_Total          227456
##  6 FY13        Oct_Total          224624
##  7 FY14        Oct_Total          206471
##  8 FY15        Oct_Total          254294
##  9 FY16        Oct_Total          244599
## 10 FY17        Oct_Total          173396
## 11 FY08        Oct_Internet        15082
## 12 FY09        Oct_Internet        32578
## 13 FY10        Oct_Internet        65533
## 14 FY11        Oct_Internet        92856
## 15 FY12        Oct_Internet        86811
## 16 FY13        Oct_Internet        92542
## 17 FY14        Oct_Internet        98400
## 18 FY15        Oct_Internet       133740
## 19 FY16        Oct_Internet       125971
## 20 FY17        Oct_Internet        90325
## # ... with 220 more rows
```

Split the month and application type

```
Social_Security_long <- separate(Social_Security_long, month, c("month",
"application_method"), sep="_")

# View result
print(Social_Security_long, n=20)
```

```
## # A tibble: 240 x 4
##    Fiscal_Year month application_method applications
##    <chr>       <chr> <chr>                     <int>
##  1 FY08        Oct   Total                    176407
##  2 FY09        Oct   Total                    244781
##  3 FY10        Oct   Total                    286598
##  4 FY11        Oct   Total                    299033
##  5 FY12        Oct   Total                    227456
##  6 FY13        Oct   Total                    224624
##  7 FY14        Oct   Total                    206471
##  8 FY15        Oct   Total                    254294
##  9 FY16        Oct   Total                    244599
## 10 FY17        Oct   Total                    173396
## 11 FY08        Oct   Internet                  15082
## 12 FY09        Oct   Internet                  32578
```

```
## 13 FY10        Oct   Internet                  65533
## 14 FY11        Oct   Internet                  92856
## 15 FY12        Oct   Internet                  86811
## 16 FY13        Oct   Internet                  92542
## 17 FY14        Oct   Internet                  98400
## 18 FY15        Oct   Internet                 133740
## 19 FY16        Oct   Internet                 125971
## 20 FY17        Oct   Internet                  90325
## # ... with 220 more rows
```

Convert month to standard abbreviations

```
# Values for months
unique(Social_Security_long$month)

##  [1] "Oct"    "Nov"    "Dec"    "Jan"    "Feb"    "Mar"    "Apr"
##  [8] "May"    "June"   "July"   "August" "Sept"

# Convert month to standard abbreviations
Social_Security_long$month <- substr(Social_Security_long$month,1,3)

# Values for months and years
unique(Social_Security_long$month)

##  [1] "Oct" "Nov" "Dec" "Jan" "Feb" "Mar" "Apr" "May" "Jun" "Jul" "Aug"
## [12] "Sep"

unique(Social_Security_long$Fiscal_Year)

##  [1] "FY08" "FY09" "FY10" "FY11" "FY12" "FY13" "FY14" "FY15" "FY16" "FY17"
```

Convert Fiscal_Year from alphanumeric strings to actual years

```
Social_Security_long$Fiscal_Year <-
str_replace(Social_Security_long$Fiscal_Year, "FY", "20")

# values for months
unique(Social_Security_long$Fiscal_Year)

##  [1] "2008" "2009" "2010" "2011" "2012" "2013" "2014" "2015" "2016" "2017"
```

Build a date string using the first day of the month

```
paste('01', Social_Security_long$month, Social_Security_long$Fiscal_Year)

##   [1] "01 Oct 2008" "01 Oct 2009" "01 Oct 2010" "01 Oct 2011" "01 Oct
2012"
##   [6] "01 Oct 2013" "01 Oct 2014" "01 Oct 2015" "01 Oct 2016" "01 Oct
2017"
##  [11] "01 Oct 2008" "01 Oct 2009" "01 Oct 2010" "01 Oct 2011" "01 Oct
2012"
##  [16] "01 Oct 2013" "01 Oct 2014" "01 Oct 2015" "01 Oct 2016" "01 Oct
2017"
##  [21] "01 Nov 2008" "01 Nov 2009" "01 Nov 2010" "01 Nov 2011" "01 Nov
2012"
```

```
##  [26] "01 Nov 2013" "01 Nov 2014" "01 Nov 2015" "01 Nov 2016" "01 Nov
2017"
##  [31] "01 Nov 2008" "01 Nov 2009" "01 Nov 2010" "01 Nov 2011" "01 Nov
2012"
##  [36] "01 Nov 2013" "01 Nov 2014" "01 Nov 2015" "01 Nov 2016" "01 Nov
2017"
##  [41] "01 Dec 2008" "01 Dec 2009" "01 Dec 2010" "01 Dec 2011" "01 Dec
2012"
##  [46] "01 Dec 2013" "01 Dec 2014" "01 Dec 2015" "01 Dec 2016" "01 Dec
2017"
##  [51] "01 Dec 2008" "01 Dec 2009" "01 Dec 2010" "01 Dec 2011" "01 Dec
2012"
##  [56] "01 Dec 2013" "01 Dec 2014" "01 Dec 2015" "01 Dec 2016" "01 Dec
2017"
##  [61] "01 Jan 2008" "01 Jan 2009" "01 Jan 2010" "01 Jan 2011" "01 Jan
2012"
##  [66] "01 Jan 2013" "01 Jan 2014" "01 Jan 2015" "01 Jan 2016" "01 Jan
2017"
##  [71] "01 Jan 2008" "01 Jan 2009" "01 Jan 2010" "01 Jan 2011" "01 Jan
2012"
##  [76] "01 Jan 2013" "01 Jan 2014" "01 Jan 2015" "01 Jan 2016" "01 Jan
2017"
##  [81] "01 Feb 2008" "01 Feb 2009" "01 Feb 2010" "01 Feb 2011" "01 Feb
2012"
##  [86] "01 Feb 2013" "01 Feb 2014" "01 Feb 2015" "01 Feb 2016" "01 Feb
2017"
##  [91] "01 Feb 2008" "01 Feb 2009" "01 Feb 2010" "01 Feb 2011" "01 Feb
2012"
##  [96] "01 Feb 2013" "01 Feb 2014" "01 Feb 2015" "01 Feb 2016" "01 Feb
2017"
## [101] "01 Mar 2008" "01 Mar 2009" "01 Mar 2010" "01 Mar 2011" "01 Mar
2012"
## [106] "01 Mar 2013" "01 Mar 2014" "01 Mar 2015" "01 Mar 2016" "01 Mar
2017"
## [111] "01 Mar 2008" "01 Mar 2009" "01 Mar 2010" "01 Mar 2011" "01 Mar
2012"
## [116] "01 Mar 2013" "01 Mar 2014" "01 Mar 2015" "01 Mar 2016" "01 Mar
2017"
## [121] "01 Apr 2008" "01 Apr 2009" "01 Apr 2010" "01 Apr 2011" "01 Apr
2012"
## [126] "01 Apr 2013" "01 Apr 2014" "01 Apr 2015" "01 Apr 2016" "01 Apr
2017"
## [131] "01 Apr 2008" "01 Apr 2009" "01 Apr 2010" "01 Apr 2011" "01 Apr
2012"
## [136] "01 Apr 2013" "01 Apr 2014" "01 Apr 2015" "01 Apr 2016" "01 Apr
2017"
## [141] "01 May 2008" "01 May 2009" "01 May 2010" "01 May 2011" "01 May
2012"
## [146] "01 May 2013" "01 May 2014" "01 May 2015" "01 May 2016" "01 May
2017"
```

```
## [151] "01 May 2008" "01 May 2009" "01 May 2010" "01 May 2011" "01 May
2012"
## [156] "01 May 2013" "01 May 2014" "01 May 2015" "01 May 2016" "01 May
2017"
## [161] "01 Jun 2008" "01 Jun 2009" "01 Jun 2010" "01 Jun 2011" "01 Jun
2012"
## [166] "01 Jun 2013" "01 Jun 2014" "01 Jun 2015" "01 Jun 2016" "01 Jun
2017"
## [171] "01 Jun 2008" "01 Jun 2009" "01 Jun 2010" "01 Jun 2011" "01 Jun
2012"
## [176] "01 Jun 2013" "01 Jun 2014" "01 Jun 2015" "01 Jun 2016" "01 Jun
2017"
## [181] "01 Jul 2008" "01 Jul 2009" "01 Jul 2010" "01 Jul 2011" "01 Jul
2012"
## [186] "01 Jul 2013" "01 Jul 2014" "01 Jul 2015" "01 Jul 2016" "01 Jul
2017"
## [191] "01 Jul 2008" "01 Jul 2009" "01 Jul 2010" "01 Jul 2011" "01 Jul
2012"
## [196] "01 Jul 2013" "01 Jul 2014" "01 Jul 2015" "01 Jul 2016" "01 Jul
2017"
## [201] "01 Aug 2008" "01 Aug 2009" "01 Aug 2010" "01 Aug 2011" "01 Aug
2012"
## [206] "01 Aug 2013" "01 Aug 2014" "01 Aug 2015" "01 Aug 2016" "01 Aug
2017"
## [211] "01 Aug 2008" "01 Aug 2009" "01 Aug 2010" "01 Aug 2011" "01 Aug
2012"
## [216] "01 Aug 2013" "01 Aug 2014" "01 Aug 2015" "01 Aug 2016" "01 Aug
2017"
## [221] "01 Sep 2008" "01 Sep 2009" "01 Sep 2010" "01 Sep 2011" "01 Sep
2012"
## [226] "01 Sep 2013" "01 Sep 2014" "01 Sep 2015" "01 Sep 2016" "01 Sep
2017"
## [231] "01 Sep 2008" "01 Sep 2009" "01 Sep 2010" "01 Sep 2011" "01 Sep
2012"
## [236] "01 Sep 2013" "01 Sep 2014" "01 Sep 2015" "01 Sep 2016" "01 Sep
2017"

Social_Security_long$Date <- dmy(paste("01", Social_Security_long$month,
Social_Security$Fiscal_Year))

# Unique values
unique(Social_Security_long$Date)

##   [1] "2008-10-01" "2009-10-01" "2010-10-01" "2011-10-01" "2012-10-01"
##   [6] "2013-10-01" "2014-10-01" "2015-10-01" "2016-10-01" "2017-10-01"
##  [11] "2008-11-01" "2009-11-01" "2010-11-01" "2011-11-01" "2012-11-01"
##  [16] "2013-11-01" "2014-11-01" "2015-11-01" "2016-11-01" "2017-11-01"
##  [21] "2008-12-01" "2009-12-01" "2010-12-01" "2011-12-01" "2012-12-01"
##  [26] "2013-12-01" "2014-12-01" "2015-12-01" "2016-12-01" "2017-12-01"
##  [31] "2008-01-01" "2009-01-01" "2010-01-01" "2011-01-01" "2012-01-01"
```

```
##  [36] "2013-01-01" "2014-01-01" "2015-01-01" "2016-01-01" "2017-01-01"
##  [41] "2008-02-01" "2009-02-01" "2010-02-01" "2011-02-01" "2012-02-01"
##  [46] "2013-02-01" "2014-02-01" "2015-02-01" "2016-02-01" "2017-02-01"
##  [51] "2008-03-01" "2009-03-01" "2010-03-01" "2011-03-01" "2012-03-01"
##  [56] "2013-03-01" "2014-03-01" "2015-03-01" "2016-03-01" "2017-03-01"
##  [61] "2008-04-01" "2009-04-01" "2010-04-01" "2011-04-01" "2012-04-01"
##  [66] "2013-04-01" "2014-04-01" "2015-04-01" "2016-04-01" "2017-04-01"
##  [71] "2008-05-01" "2009-05-01" "2010-05-01" "2011-05-01" "2012-05-01"
##  [76] "2013-05-01" "2014-05-01" "2015-05-01" "2016-05-01" "2017-05-01"
##  [81] "2008-06-01" "2009-06-01" "2010-06-01" "2011-06-01" "2012-06-01"
##  [86] "2013-06-01" "2014-06-01" "2015-06-01" "2016-06-01" "2017-06-01"
##  [91] "2008-07-01" "2009-07-01" "2010-07-01" "2011-07-01" "2012-07-01"
##  [96] "2013-07-01" "2014-07-01" "2015-07-01" "2016-07-01" "2017-07-01"
## [101] "2008-08-01" "2009-08-01" "2010-08-01" "2011-08-01" "2012-08-01"
## [106] "2013-08-01" "2014-08-01" "2015-08-01" "2016-08-01" "2017-08-01"
## [111] "2008-09-01" "2009-09-01" "2010-09-01" "2011-09-01" "2012-09-01"
## [116] "2013-09-01" "2014-09-01" "2015-09-01" "2016-09-01" "2017-09-01"
```

- Government fiscal years differ from calendar years in that they are named for the calendar year where they end. The government fiscal year begins in October.
- Thus, October 2016 is actually in FY17.
- We must convert these values to calendar dates before we try to plot them, so we need to find months >=10 and subtract one year from them

```
# Looking for affected rows
advanced_dates <- which(month(Social_Security_long$Date)>=10)

# Decerement the years by one
year(Social_Security_long$Date[advanced_dates]) <-
year(Social_Security_long$Date[advanced_dates]) - 1

# View Summary of dataset
summary(Social_Security_long)
```

```
##  Fiscal_Year           month           application_method  applications
##  Length:240         Length:240         Length:240         Min.   : 14321
##  Class :character   Class :character   Class :character   1st Qu.: 91399
##  Mode  :character   Mode  :character   Mode  :character   Median :145344
##                                                           Mean   :154322
##                                                           3rd Qu.:224669
##                                                           Max.   :300442
##                                                           NA's   :16
##       Date
##  Min.   :2007-10-01
##  1st Qu.:2010-03-24
##  Median :2012-09-16
##  Mean   :2012-09-15
##  3rd Qu.:2015-03-08
##  Max.   :2017-09-01
##
```

```r
# Remove Fiscal_Year and month columns
Social_Security_long$Fiscal_Year <- NULL
Social_Security_long$month <- NULL

# Convert application_method to a factor
Social_Security_long$application_method <-
as.factor(Social_Security_long$application_method)

# Summary of dataset
summary(Social_Security_long)
```

```
##  application_method  applications         Date
##  Internet:120       Min.   : 14321   Min.   :2007-10-01
##  Total   :120       1st Qu.: 91399   1st Qu.:2010-03-24
##                     Median :145344   Median :2012-09-16
##                     Mean   :154322   Mean   :2012-09-15
##                     3rd Qu.:224669   3rd Qu.:2015-03-08
##                     Max.   :300442   Max.   :2017-09-01
##                     NA's   :16
```

```r
# Widen the final dataset
Social_Security <- spread(Social_Security_long, application_method,
applications)

# View result
print(Social_Security,n=20)
```

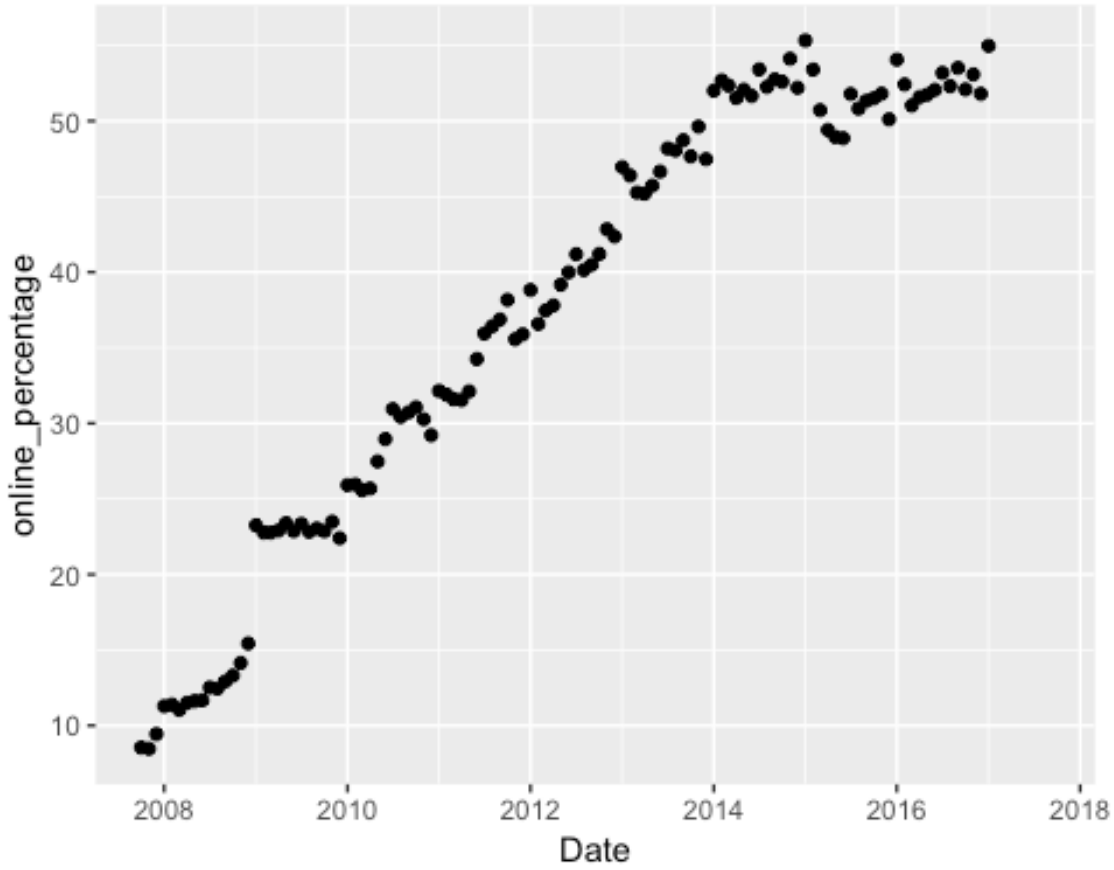
```
## # A tibble: 120 x 3
##    Date       Internet  Total
##    <date>        <int>  <int>
##  1 2007-10-01    15082 176407
##  2 2007-11-01    17301 204287
##  3 2007-12-01    14321 151687
##  4 2008-01-01    18391 162966
##  5 2008-02-01    26034 228623
##  6 2008-03-01    21064 190716
##  7 2008-04-01    22372 194403
##  8 2008-05-01    26337 226549
##  9 2008-06-01    22551 193094
## 10 2008-07-01    22728 181552
## 11 2008-08-01    30580 245429
## 12 2008-09-01    24141 186750
## 13 2008-10-01    32578 244781
## 14 2008-11-01    25620 181161
## 15 2008-12-01    27174 176107
## 16 2009-01-01    57908 249062
## 17 2009-02-01    50408 221368
## 18 2009-03-01    53592 235360
## 19 2009-04-01    53675 234304
```

```
## 20 2009-05-01    65822 281343
## # ... with 100 more rows
```

Plot percentage of applicats that were online

```
# Add a column to see the percentage of applicats that were online
Social_Security$online_percentage <-
Social_Security$Internet/Social_Security$Total*100

# Plot the results
ggplot(data=Social_Security, mapping=aes(x=Date,y=online_percentage)) +
  geom_point()
```



Write clean CSV in R

```
write.csv(Social_Security, file =
"Social_Security_Disability_final_version.csv")
```