Assignment 4

Senhao Li

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------------- tidyverse 1.2.1 --
```

```
## √ ggplot2 2.2.1     √ purrr   0.2.4
## √ tibble  1.4.2     √ dplyr   0.7.4
## √ tidyr   0.8.0     √ stringr 1.2.0
## √ readr   1.1.1     √ forcats 0.2.0
```

```
## -- Conflicts ------------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

10.5 exercise

1.How can you tell if an object is a tibble?

```
head(mtcars)
```

```
##                    mpg cyl disp  hp drat    wt  qsec vs am gear carb
## Mazda RX4         21.0   6  160 110 3.90 2.620 16.46  0  1    4    4
## Mazda RX4 Wag     21.0   6  160 110 3.90 2.875 17.02  0  1    4    4
## Datsun 710        22.8   4  108  93 3.85 2.320 18.61  1  1    4    1
## Hornet 4 Drive    21.4   6  258 110 3.08 3.215 19.44  1  0    3    1
## Hornet Sportabout 18.7   8  360 175 3.15 3.440 17.02  0  0    3    2
## Valiant           18.1   6  225 105 2.76 3.460 20.22  1  0    3    1
```

```
is.tibble(mtcars)
```

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

```
class(mtcars)
```

```
## [1] "data.frame"
```

2.Compare and contrast the following operations on a data.frame and equivalent tibble.

```
df <- data.frame(abc = 1, xyz = "a")
df
```

```
##   abc xyz
## 1   1   a
```

```
df$x #the name of the column in data frame can be automatically completed and recognized by R
```

```
## [1] a
## Levels: a
```

```r
class(df[, "xyz"]) #it returns a factor if one single value is seleted in a data frame
```

```
## [1] "factor"
```

```r
class(df[, c("abc", "xyz")])
```

```
## [1] "data.frame"
```

```r
tb <- as_tibble(df)
tb
```

```
## # A tibble: 1 x 2
##     abc xyz
##   <dbl> <fct>
## 1  1.00 a
```

```r
tb$x #opposed to what happened to data frame, the incomplete name of the column of a tibble cannot be r
```

```
## Warning: Unknown or uninitialised column: 'x'.
```

```
## NULL
```

```r
class(tb[, "xyz"]) # it returns a datafrane even if only a single value is selected.
```

```
## [1] "tbl_df"     "tbl"        "data.frame"
```

```r
class(tb[, c("abc", "xyz")])
```

```
## [1] "tbl_df"     "tbl"        "data.frame"
```

3. how can you extract the reference variable from a tibble?

```r
df3 <- tibble(a="mpg",b=23)
df3[["a"]]
```

```
## [1] "mpg"
```

4.

```r
annoying <- tibble(
  `1` = 1:10,
  `2` = `1` * 2 + rnorm(length(`1`))
)
```

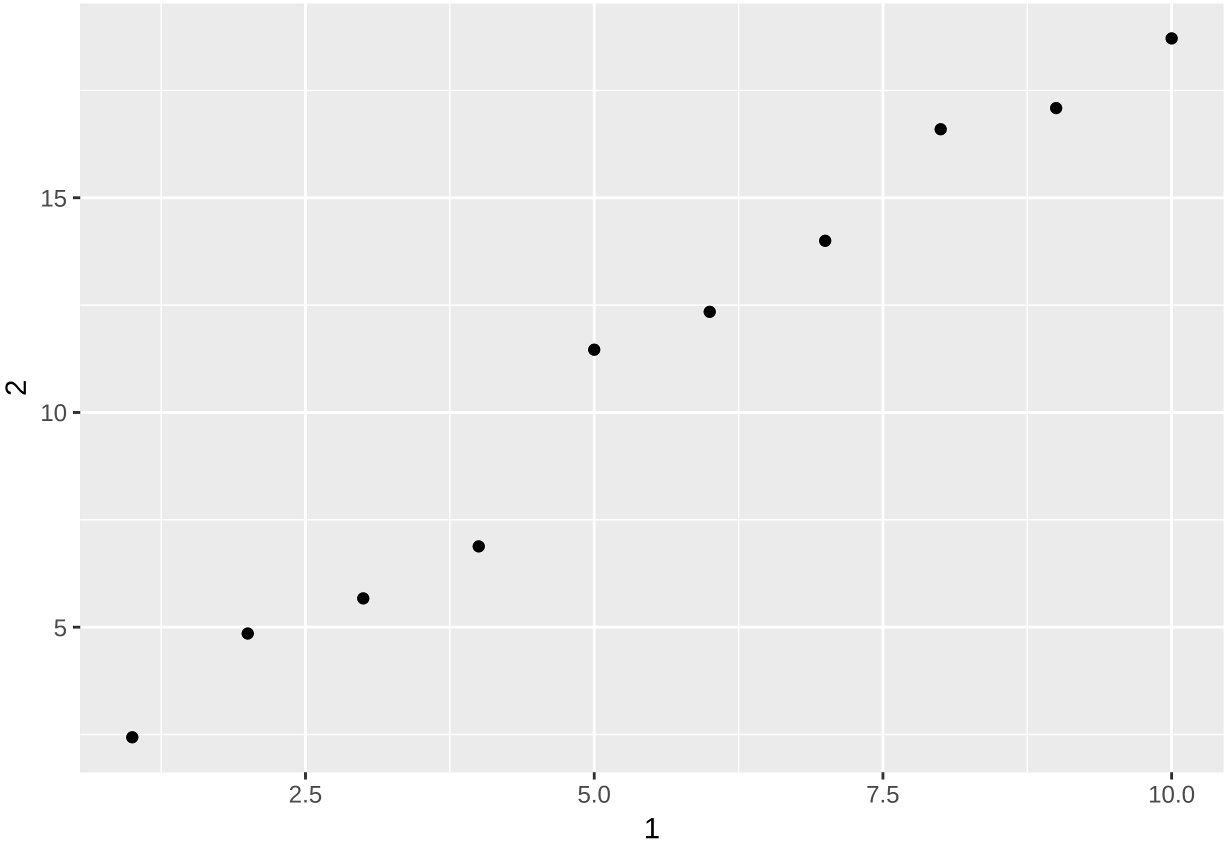
1)

```r
annoying[["1"]]
```

```
##  [1]  1  2  3  4  5  6  7  8  9 10
```

2)

```
ggplot(annoying, aes(x = `1`, y = `2`)) +
  geom_point()
```



#3)

```
annoying_new <- tibble(
  `1` = 1:10,
  `2` = `1` * 2 + rnorm(length(`1`)),
  `3` = `2`/`1`
)
```

4)

```
names(annoying_new) <- c("one","two","three")
annoying_new
```

```
## # A tibble: 10 x 3
##       one   two three
##     <int> <dbl> <dbl>
##  1      1  1.78  1.78
##  2      2  4.09  2.05
##  3      3  6.87  2.29
##  4      4  9.27  2.32
##  5     5  9.32  1.86
##  6     6 11.7   1.96
##  7     7 12.6   1.80
##  8     8 16.1   2.01
##  9     9 17.8   1.98
## 10    10 20.8   2.08
```

```
#5.
enframe(c(a = 14, b = 12, c = 45))
```

```
## # A tibble: 3 x 2
##   name  value
##   <chr> <dbl>
## 1 a      14.0
## 2 b      12.0
## 3 c      45.0
```

```
# it makes named vectors a data frame with name and value
```

```
#6.
#n_extra in print.tbl_df
```

12.6.1

1.

```
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
```

```
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
```

```
who1 %>%
  filter(cases == 0) %>%
  nrow()
```

```
## [1] 11080
```

```
gather(who, new_sp_m014:newrel_f65, key = "key", value = "cases") %>%
  group_by(country, year)  %>%
  mutate(missing = is.na(cases)) %>%
  select(country, year, missing) %>%
  distinct() %>%
  group_by(country, year) %>%
  filter(n() > 1)
```

```
## # A tibble: 6,968 x 3
## # Groups:   country, year [3,484]
##    country      year missing
##    <chr>       <int> <lgl>
##  1 Afghanistan  1997 F
##  2 Afghanistan  1998 F
##  3 Afghanistan  1999 F
##  4 Afghanistan  2000 F
##  5 Afghanistan  2001 F
##  6 Afghanistan  2002 F
##  7 Afghanistan  2003 F
##  8 Afghanistan  2004 F
##  9 Afghanistan  2005 F
## 10 Afghanistan  2006 F
## # ... with 6,958 more rows
```

```
#2.
who3a <- who1 %>%
  separate(key, c("new", "type", "sexage"), sep = "_")
```

```
## Warning: Expected 3 pieces. Missing pieces filled with `NA` in 2580 rows
## [73467, 73468, 73469, 73470, 73471, 73472, 73473, 73474, 73475, 73476,
## 73477, 73478, 73479, 73480, 73481, 73482, 73483, 73484, 73485, 73486, ...].
```

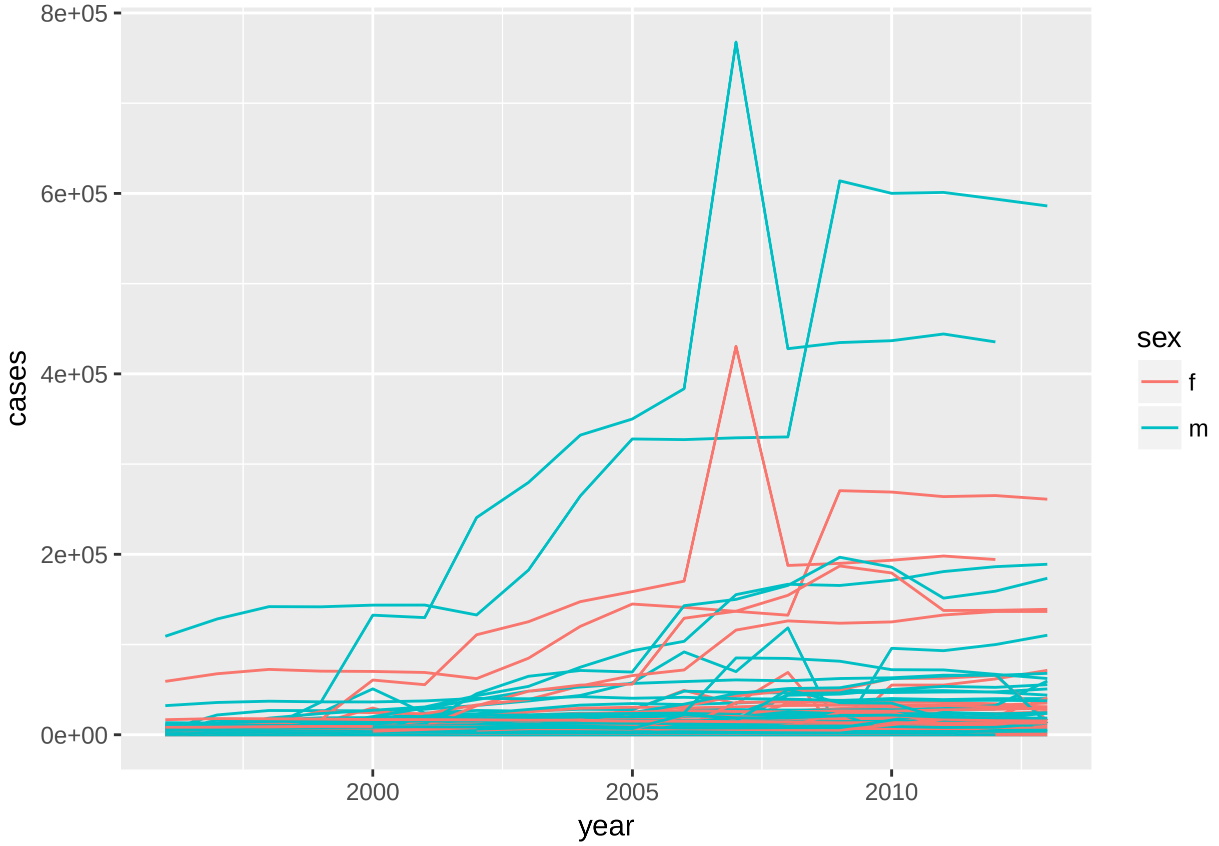
```
filter(who3a, new == "newrel") %>% head()
```

```
## # A tibble: 6 x 8
##   country     iso2  iso3   year new    type  sexage cases
##   <chr>       <chr> <chr> <int> <chr>  <chr> <chr>  <int>
## 1 Afghanistan AF    AFG    2013 newrel m014  <NA>    1705
## 2 Albania     AL    ALB    2013 newrel m014  <NA>      14
## 3 Algeria     DZ    DZA    2013 newrel m014  <NA>      25
## 4 Andorra     AD    AND    2013 newrel m014  <NA>       0
## 5 Angola      AO    AGO    2013 newrel m014  <NA>     486
## 6 Anguilla    AI    AIA    2013 newrel m014  <NA>       0
```

```
#3.
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.
who5 %>%
  group_by(country, year, sex) %>%
  filter(year > 1995) %>%
  summarise(cases = sum(cases)) %>%
  unite(country_sex, country, sex, remove = FALSE) %>%
  ggplot(aes(x = year, y = cases, group = country_sex, colour = sex)) +
  geom_line()
```



```
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
```

```
#table 4 to 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
```

```r
library(reshape2)
```

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

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

```r
source("xtable.r")

# Data from http://pewforum.org/Datasets/Dataset-Download.aspx

# Load data -----------------------------------------------------------------

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
```

```r
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/refu

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

```
##   religion  income freq
## 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
```

```
xtable(counts[1:10, ], file = "pew-clean.tex")

# 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.
```

```
xtable(raw[1:10, 1:7], file = "pew-raw.tex")
head(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
##   $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
```

```
table6 <- gather(raw, income,freq,"<$10k", "$10-20k", "$20-30k", "$30-40k", "$40-50k", "$50-75k", "$75-
table6 <- arrange(table6,religion)
head(table6)
```

```
##   religion  income freq
## 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
```

```
#table 7 to table 8
options(stringsAsFactors = FALSE)
library(lubridate)
```

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

```
## The following object is masked from 'package:plyr':
##
##     here
```

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

```
library(reshape2)
library(stringr)
library(plyr)
source("xtable.r")

raw <- read.csv("billboard.csv")
raw <- raw[, c("year", "artist.inverted", "track", "time", "date.entered", "x1st.week", "x2nd.week", "x3
names(raw)[2] <- "artist"

raw$artist <- iconv(raw$artist, "MAC", "ASCII//translit")
raw$track <- str_replace(raw$track, " \\(.*?\\)", "")
names(raw)[-(1:5)] <- str_c("wk", 1:76)
raw <- arrange(raw, year, artist, track)

long_name <- nchar(raw$track) > 20
raw$track[long_name] <- paste0(substr(raw$track[long_name], 0, 20), "...")

xtable(raw[c(1:3, 6:10), 1:8], "billboard-raw.tex")

table8 <- gather(raw,key="week", value="rank",str_c("wk", 1:76))
table8 <- arrange(table8,artist) %>%
  na.omit(table8)
table8$week <- str_replace(table8$week,"wk","")
head(table8)
```

```
##   year artist           track time date.entered week rank
## 1 2000   2 Pac Baby Don't Cry 4:22   2000-02-26    1   87
## 2 2000   2 Pac Baby Don't Cry 4:22   2000-02-26    2   82
## 3 2000   2 Pac Baby Don't Cry 4:22   2000-02-26    3   72
## 4 2000   2 Pac Baby Don't Cry 4:22   2000-02-26    4   77
## 5 2000   2 Pac Baby Don't Cry 4:22   2000-02-26    5   87
## 6 2000   2 Pac Baby Don't Cry 4:22   2000-02-26    6   94
```