

Exercises 3

Nick Gembs

2/6/2023

Stat Computing - Exercises 03 - Billboard Charts

On canvas, there is a file called `Hot_100.csv`, which contains data from the Billboard 100 music charts going back to 1958. Below are some exercises on functions and subsetting that make use of the data. For all questions, include 3 examples to demonstrate calling your function with different arguments.

```
dat = read.csv("C:/Users/Nick/Downloads/Hot_100.csv")
head(dat)
```

```
##   chart_position chart_date      song      performer
## 1             84 1990-05-05 "B" Girls Young And Restless
## 2             78 1990-05-12 "B" Girls Young And Restless
## 3             68 1990-05-19 "B" Girls Young And Restless
## 4             60 1990-05-26 "B" Girls Young And Restless
## 5             58 1990-06-02 "B" Girls Young And Restless
## 6             55 1990-06-09 "B" Girls Young And Restless
##               song_id instance time_on_chart consecutive_weeks
## 1 "B" GirlsYoung And Restless      1          1              NA
## 2 "B" GirlsYoung And Restless      1          2              1
## 3 "B" GirlsYoung And Restless      1          3              2
## 4 "B" GirlsYoung And Restless      1          4              3
## 5 "B" GirlsYoung And Restless      1          5              4
## 6 "B" GirlsYoung And Restless      1          6              5
##   previous_week peak_position worst_position chart_debut
## 1           NA           84           84 1990-05-05
## 2           84           78           84 1990-05-05
## 3           78           68           84 1990-05-05
## 4           68           60           84 1990-05-05
## 5           60           58           84 1990-05-05
## 6           58           55           84 1990-05-05
##               chart_url
## 1 https://www.billboard.com/charts/hot-100/1990-05-05
## 2 https://www.billboard.com/charts/hot-100/1990-05-12
## 3 https://www.billboard.com/charts/hot-100/1990-05-19
## 4 https://www.billboard.com/charts/hot-100/1990-05-26
## 5 https://www.billboard.com/charts/hot-100/1990-06-02
## 6 https://www.billboard.com/charts/hot-100/1990-06-09
```

1. Write a function that takes in a character string indicating a date, an optional format argument that has a default date format. The function should return the #1 song and its performer during the week

that contains the date argument. Assume that the date in the dataset is the date of the beginning of the week. Your function should use the default date format if a format is not provided, and it should use the user-supplied format if it is supplied. Test your function with a few different dates and different formats.

```
topsong <- function(date, format = "%Y-%m-%d") {
  date = as.Date(date, format = format)
  weekdaychange = 6 - as.numeric(format(date, "%u"))

  date = format(date+weekdaychange, "%Y-%m-%d")

  return(dat[(dat$chart_date == date) & (dat$chart_position == 1),c("song","performer")])
}

topsong("1990-05-05")
```

```
##              song              performer
## 203116 Nothing Compares 2 U Sinead O'Connor
```

```
topsong("1990-07-17" , format = "%Y-%m-%d")
```

```
##              song              performer
## 241183 She Ain't Worth It Glenn Medeiros Featuring Bobby Brown
```

```
topsong("1990/07/17" , format = "%Y/%m/%d")
```

```
##              song              performer
## 241183 She Ain't Worth It Glenn Medeiros Featuring Bobby Brown
```

```
topsong("17-07-1990" , format = "%d-%m-%Y")
```

```
##              song              performer
## 241183 She Ain't Worth It Glenn Medeiros Featuring Bobby Brown
```

```
topsong("28-09-1990" , format = "%d-%m-%Y")
```

```
##              song performer
## 449 (Can't Live Without Your) Love And Affection Nelson
```

2. Modify your function to add a `top` argument, indicating that the function should return the top `top` songs on the specified date. Make sure that the returned songs are listed in order according to the weekly ranking. Test your function by printing the top 10 songs on the day you were born.

```
topsong <- function(date, format = "%Y-%m-%d", top = 1) {
  date = as.Date(date, format = format)
  weekdaychange = 6 - as.numeric(format(date, "%u"))

  date = format(date+weekdaychange, "%Y-%m-%d")
```

```

result = (dat[(dat$chart_date == date) & (dat$chart_position <= top),c("song", "performer", "chart_posi

return(result[order(result$chart_position),c("song", "performer")])
}

topsong("2002-07-17", top = 10)

```

```

##              song
## 116086      Hot In Herre
## 323731      Without Me
## 110455      Hero
## 52114      Complicated
## 128869 I Need A Girl (Part Two)
## 204531      Oh Boy
## 280667      The Middle
## 7465      A Thousand Miles
## 86333      Foolish
## 137034      I'm Gonna Be Alright
##
##              performer
## 116086      Nelly
## 323731      Eminem
## 110455      Chad Kroeger Featuring Josey Scott
## 52114      Avril Lavigne
## 128869 P. Diddy & Ginuwine Featuring Loon, Mario Winans & Tammy Ruggeri
## 204531      Cam'Ron Featuring Juelz Santana
## 280667      Jimmy Eat World
## 7465      Vanessa Carlton
## 86333      Ashanti
## 137034      Jennifer Lopez Featuring Nas

```

```

topsong("2002-2-11", top = 5)

```

```

##              song              performer
## 300515      U Got It Bad      Usher
## 118526 How You Remind Me      Nickelback
## 14747      Always On Time      Ja Rule Featuring Ashanti
## 9831      Ain't It Funny Jennifer Lopez Featuring Ja Rule
## 144231      In The End      Linkin Park

```

```

topsong("2022-2-10", top = 1)

```

```

##              song
## 309010 We Don't Talk About Bruno
##
## 309010 Carolina Gaitan, Mauro Castillo, Adassa, Rhenzy Feliz, Diane Guerrero, Stephanie Beatriz & En

```

```

topsong("2002-1-28", top = 5)

```

```

##              song              performer
## 300513      U Got It Bad      Usher

```

```
## 118524      How You Remind Me      Nickelback
## 14745      Always On Time Ja Rule Featuring Ashanti
## 92673      Get The Party Started      P!nk
## 194746      My Sacrifice      Creed
```

- Write a function to search for matching performer names. The function should take in a character string as its argument, find all unique performer names that contain the argument, and return them. Include an argument to determine whether or not to do exact matching on uppercase vs. lowercase. It may be a good idea to look at the documentation `grep`, `tolower`, and `toupper`.

```
SearchArtist <- function(artist, MatchCase = FALSE){
  if (MatchCase == FALSE){
    return ( dat[grep(artist, dat$performer, ignore.case = TRUE, fixed = FALSE),])
  } else if (MatchCase == TRUE) {
    return ( dat[grep(artist, dat$performer, ignore.case = FALSE, fixed = TRUE),])
  }
}

head(SearchArtist("Emin"))
```

```
##      chart_position chart_date      song performer      song_id
## 3949           92 2010-07-10    25 To Life    Eminem    25 To LifeEminem
## 4160           32 2009-05-16      3am      Eminem      3amEminem
## 19812          83 2005-06-18 Ass Like That    Eminem Ass Like ThatEminem
## 19813          71 2005-06-25 Ass Like That    Eminem Ass Like ThatEminem
## 19814          60 2005-07-02 Ass Like That    Eminem Ass Like ThatEminem
## 19815          67 2005-07-09 Ass Like That    Eminem Ass Like ThatEminem
##      instance time_on_chart consecutive_weeks previous_week peak_position
## 3949          1           1           NA           NA           92
## 4160          1           1           NA           NA           32
## 19812         1           1           NA           NA           83
## 19813         1           2            1           83           71
## 19814         1           3            2           71           60
## 19815         1           4            3           60           60
##      worst_position chart_debut
## 3949           92 2010-07-10
## 4160           32 2009-05-16
## 19812          83 2005-06-18
## 19813          83 2005-06-18
## 19814          83 2005-06-18
## 19815          83 2005-06-18
##
##      chart_url
## 3949 https://www.billboard.com/charts/hot-100/2010-07-10
## 4160 https://www.billboard.com/charts/hot-100/2009-05-16
## 19812 https://www.billboard.com/charts/hot-100/2005-06-18
## 19813 https://www.billboard.com/charts/hot-100/2005-06-25
## 19814 https://www.billboard.com/charts/hot-100/2005-07-02
## 19815 https://www.billboard.com/charts/hot-100/2005-07-09
```

```
head(SearchArtist(("Charlie Puth"), MatchCase = TRUE))
```

```
##      chart_position chart_date      song      performer      song_id
```

```
## 20140      68 2017-05-13 Attention Charlie Puth AttentionCharlie Puth
## 20141      64 2017-05-20 Attention Charlie Puth AttentionCharlie Puth
## 20142      53 2017-05-27 Attention Charlie Puth AttentionCharlie Puth
## 20143      45 2017-06-03 Attention Charlie Puth AttentionCharlie Puth
## 20144      35 2017-06-10 Attention Charlie Puth AttentionCharlie Puth
## 20145      37 2017-06-17 Attention Charlie Puth AttentionCharlie Puth
##      instance time_on_chart consecutive_weeks previous_week peak_position
## 20140          1              1              NA              NA              68
## 20141          1              2              1              68              64
## 20142          1              3              2              64              53
## 20143          1              4              3              53              45
## 20144          1              5              4              45              35
## 20145          1              6              5              35              35
##      worst_position chart_debut
## 20140          68 2017-05-13
## 20141          68 2017-05-13
## 20142          68 2017-05-13
## 20143          68 2017-05-13
## 20144          68 2017-05-13
## 20145          68 2017-05-13
##
##      chart_url
## 20140 https://www.billboard.com/charts/hot-100/2017-05-13
## 20141 https://www.billboard.com/charts/hot-100/2017-05-20
## 20142 https://www.billboard.com/charts/hot-100/2017-05-27
## 20143 https://www.billboard.com/charts/hot-100/2017-06-03
## 20144 https://www.billboard.com/charts/hot-100/2017-06-10
## 20145 https://www.billboard.com/charts/hot-100/2017-06-17
```

```
head(SearchArtist("Joe"), MatchCase = TRUE))
```

```
##      chart_position chart_date      song
## 1307          94 1964-01-25      (It's No) Sin
## 1308          82 1964-02-01      (It's No) Sin
## 1309          74 1964-02-08      (It's No) Sin
## 1310          88 1964-02-15      (It's No) Sin
## 1311          84 1964-02-22      (It's No) Sin
## 2303          88 1968-04-13 (You Keep Me) Hangin' On
##
##      performer
## 1307 The Duprees featuring Joey Vann
## 1308 The Duprees featuring Joey Vann
## 1309 The Duprees featuring Joey Vann
## 1310 The Duprees featuring Joey Vann
## 1311 The Duprees featuring Joey Vann
## 2303      Joe Simon
##
##      song_id instance time_on_chart
## 1307 (It's No) SinThe Duprees featuring Joey Vann      1      1
## 1308 (It's No) SinThe Duprees featuring Joey Vann      1      2
## 1309 (It's No) SinThe Duprees featuring Joey Vann      1      3
## 1310 (It's No) SinThe Duprees featuring Joey Vann      1      4
## 1311 (It's No) SinThe Duprees featuring Joey Vann      1      5
## 2303      (You Keep Me) Hangin' OnJoe Simon      1      1
##
##      consecutive_weeks previous_week peak_position worst_position chart_debut
## 1307      NA      NA      94      94 1964-01-25
## 1308          1      94      82      94 1964-01-25
```

```
## 1309      2      82      74      94 1964-01-25
## 1310      3      74      74      94 1964-01-25
## 1311      4      88      74      94 1964-01-25
## 2303     NA      NA      88      88 1968-04-13
##                                     chart_url
## 1307 https://www.billboard.com/charts/hot-100/1964-01-25
## 1308 https://www.billboard.com/charts/hot-100/1964-02-01
## 1309 https://www.billboard.com/charts/hot-100/1964-02-08
## 1310 https://www.billboard.com/charts/hot-100/1964-02-15
## 1311 https://www.billboard.com/charts/hot-100/1964-02-22
## 2303 https://www.billboard.com/charts/hot-100/1968-04-13
```

4. Write a function to take in a performer name, and return all of the songs belonging to that performer that appeared on the Billboard list. Use exact matching for performer names.

```
artistsongs <- function (artist) {
  return ( unique(dat[grep(artist, dat$performer, ignore.case = FALSE, fixed = TRUE), "song"]))
}
```

```
artistsongs("Black Eyed Peas")
```

```
## [1] "Boom Boom Pow"      "Don't Lie"
## [3] "Don't Phunk With My Heart" "Don't Stop The Party"
## [5] "Girl Like Me"       "Hey Mama"
## [7] "I Gotta Feeling"     "Imma Be"
## [9] "Just Can't Get Enough" "Let's Get It Started"
## [11] "Mamacita"           "Meet Me Halfway"
## [13] "My Humps"           "Pump It"
## [15] "RITMO (Bad Boys For Life)" "Request Line"
## [17] "Rock That Body"     "The Time (Dirty Bit)"
## [19] "Where Is The Love?"
```

```
cat("\n")
```

```
artistsongs("Yung Gravy")
```

```
## [1] "Betty (Get Money)"
```

```
cat("\n")
```

```
artistsongs("Foo Fighters")
```

```
## [1] "All My Life"      "Best Of You"      "DOA"
## [4] "Learn To Fly"     "Long Road To Ruin" "Rope"
## [7] "The Pretender"    "Times Like These" "Walk"
## [10] "Wheels"
```

5. Song statistics: write a function that takes in a song id, and returns the following statistics for the song: number of weeks in the top 100, number of weeks in the top 40, number of weeks in the top 10, number of weeks at #1, lowest position (i.e. best position). The return type should be a data frame with columns for the song name, the performer, the first week it appeared on the charts, and all of the statistics.

```

Song_Statistics <- function (song_ID) {

  song_dat = dat[dat$song_id == song_ID,]
  top_100 = length(song_dat$song)
  top_40 = length(song_dat[song_dat$chart_position <= 40 , "song"])
  top_10 = length(song_dat[song_dat$chart_position <= 10 , "song"])
  top_1 = length(song_dat[song_dat$chart_position <= 1 , "song"])
  top = min(song_dat$chart_position)

  result = data.frame(song = song_dat[1,"song"],
                      performer = song_dat[1,"performer"],
                      first_week = song_dat[1,"chart_date"] ,
                      top_100 = top_100, top_40 = top_40 , top_10 = top_10, top_1 = top_1,
                      best = top)

  return ((result))

}

```

```
Song_Statistics("PepasFarruko")
```

```

##      song performer first_week top_100 top_40 top_10 top_1 best
## 1 Pepas   Farruko 2021-08-07      21     12      0      0    25

```

```
Song_Statistics("HelloAdele")
```

```

##      song performer first_week top_100 top_40 top_10 top_1 best
## 1 Hello     Adele 2015-11-14      26     26     18     10     1

```

```
Song_Statistics("Blinding LightsThe Weeknd")
```

```

##      song performer first_week top_100 top_40 top_10 top_1 best
## 1 Blinding Lights The Weeknd 2019-12-14     90     86     57      4     1

```

6. Song rankings: write a function that ranks all of the songs according to the statistics in the previous question. The function should take a statistic name as its argument, and return a data frame whose first column is the song name, second column is the performer, third column is the week the song first appeared, and the fourth column is the statistic. The data frame should be ordered with the best value of the statistic at the top of the data frame. Print out the rankings of the top 10 songs for 3 of the statistics.

Citation: Idea for code of this problem came from office hours

```

Rankings = function(statistic) {
  songs_id <- unique(dat$song_id)

  df2 = data.frame(colnames(c("song", "performer", "chart_debut", "top_100", "top_40", "top_10", "top_1",
                              "best")))

  for ( i in seq_along(songs_id)) {
    stats2 <- Song_Statistics(songs_id[i])
    df2 <- rbind(df2, stats2)
  }
}

```

```

}

if (statistic == "best") {
  resultdf <- (df2[order(df2[,statistic], decreasing = FALSE),c("song", "performer", "first_week", s
} else {
  resultdf <- (df2[order(df2[,statistic], decreasing = TRUE),c("song", "performer", "first_week", st
}

finalresult <- head(resultdf, 10)
return ( finalresult )

}

```

Rankings("top_40")

```

##              song              performer
## 2986      Blinding Lights      The Weeknd
## 9746          Heat Waves      Glass Animals
## 21408      Save Your Tears  The Weeknd & Ariana Grande
## 20220      Radioactive      Imagine Dragons
## 23414          Stay  The Kid LAROI & Justin Bieber
## 12544      I'm Yours      Jason Mraz
## 4345          Circles      Post Malone
## 10592      How Do I Live      LeAnn Rimes
## 7738 Foolish Games/You Were Meant For Me      Jewel
## 21848      Shape Of You      Ed Sheeran
##      first_week top_40
## 2986 2019-12-14      86
## 9746 2021-01-16      76
## 21408 2021-01-09      64
## 20220 2012-08-18      63
## 23414 2021-07-24      63
## 12544 2008-05-03      62
## 4345 2019-09-14      61
## 10592 1997-06-21      61
## 7738 1996-11-30      60
## 21848 2017-01-28      58

```

Rankings("top_1")

```

##              song
## 18650      Old Town Road
## 5507          Despacito
## 18906      One Sweet Day
## 1794          As It Was
## 3924 Candle In The Wind 1997/Something About The Way You Look Tonight
## 11299          I Gotta Feeling
## 11942      I Will Always Love You
## 12162      I'll Make Love To You
## 16278      Macarena (Bayside Boys Mix)
## 27480      Uptown Funk!
##      performer first_week top_1

```


## 18650	Lil Nas X Featuring Billy Ray Cyrus	2019-03-16	19
## 5507	Luis Fonsi & Daddy Yankee Featuring Justin Bieber	2017-02-04	16
## 18906	Mariah Carey & Boyz II Men	1995-12-02	16
## 1794	Harry Styles	2022-04-16	15
## 3924	Elton John	1997-10-11	14
## 11299	The Black Eyed Peas	2009-06-27	14
## 11942	Whitney Houston	1992-11-14	14
## 12162	Boyz II Men	1994-08-13	14
## 16278	Los Del Rio	1995-09-02	14
## 27480	Mark Ronson Featuring Bruno Mars	2014-11-29	14

Rankings("best")

##	song	performer	first_week	best
## 41	(Can't Live Without Your) Love And Affection			
## 56	(Everything I Do) I Do It For You			
## 67	(Hey Won't You Play) Another Somebody Done Somebody Wrong Song			
## 76	(I Can't Get No) Satisfaction			
## 84	(I Just) Died In Your Arms			
## 110	(I've Had) The Time Of My Life			
## 128	(Just Like) Starting Over			
## 136	(Love Is) Thicker Than Water			
## 162	(Shake, Shake, Shake) Shake Your Booty			
## 171	(Sittin' On) The Dock Of The Bay			
##		performer	first_week	best
## 41	Nelson	1990-07-07	1	
## 56	Bryan Adams	1991-06-29	1	
## 67	B.J. Thomas	1975-02-01	1	
## 76	The Rolling Stones	1965-06-12	1	
## 84	Cutting Crew	1987-03-07	1	
## 110	Bill Medley & Jennifer Warnes	1987-09-26	1	
## 128	John Lennon	1980-11-01	1	
## 136	Andy Gibb	1977-11-05	1	
## 162	KC And The Sunshine Band	1976-07-10	1	
## 171	Otis Redding	1968-01-27	1	