

Project 13 -- SEYI OGUNODEDE

Instructor: Dr. Mark Daniel

- Help with figuring out with videos and how to write a function.

Collaboration: Friend1, Friend2

Question 1

```
In [2]: file.info("/anvil/projects/tdm/data/beer/beers.csv")
# Here you can put the name of the file and get
# all kinds of information about the file
```

A data.frame: 1 x 10

	size	isdir	mode	mtime	ctime	atime	.
	<dbl>	<lgl>	<octmode>	<dttm>	<dttm>	<dttm>	.
/anvil/projects/tdm/data/beer/beers.csv	43399004	FALSE	644	2019-10-09 01:12:02	2022-07-07 21:26:23	2022-12-03 14:27:29	709

```
In [3]: file.info("/anvil/projects/tdm/data/beer/beers.csv")$size
# to know the particular quantity of the size in the column
```

43399004

```
In [4]: library(data.table)
```

```
In [5]: drinks <- fread("/anvil/projects/tdm/data/beer/beers.csv")
```

```
In [6]: dim(drinks)
# 358873 rows, 10 columns
```

358873 · 10

```
In [7]: # to Look at type of data and information about the file
str(drinks)
```

```
Classes 'data.table' and 'data.frame': 358873 obs. of 10 variables:
 $ id           : int 202522 82352 214879 320009 246438 8036 108605 345382 255286 295
56 ...
 $ name         : chr "Olde Cogitator" "Konrads Stout Russian Imperial Stout" "Scottish Right" "MegaMeow Imperial Stout" ...
 $ brewery_id   : int 2199 18604 44306 4378 44617 3469 22598 45567 11203 8203 ...
 $ state        : chr "CA" "" "IN" "WA" ...
 $ country      : chr "US" "NO" "US" "US" ...
 $ style         : chr "English Oatmeal Stout" "Russian Imperial Stout" "Scottish Ale" "American Imperial Stout" ...
 $ availability: chr "Rotating" "Rotating" "Year-round" "Winter" ...
 $ abv          : num 7.3 10.4 4 8.7 5.1 5.5 5.6 6.5 4.3 7.5 ...
 $ notes         : chr "No notes at this time." "No notes at this time." "No notes at this time." "Every time this year" ...
 $ retired       : chr "f" "f" "t" "f" ...
 - attr(*, ".internal.selfref")=<externalptr>
```

In [8]: `head(drinks)`

A data.table: 6 x 10

id	name	brewery_id	state	country	style	availability	abv	notes	retired
<int>	<chr>	<int>	<chr>	<chr>	<chr>	<chr>	<dbl>	<chr>	<chr>
202522	Olde Cogitator	2199	CA	US	English Oatmeal Stout	Rotating	7.3	No notes at this time.	f
82352	Konrads Stout Russian Imperial Stout	18604		NO	Russian Imperial Stout	Rotating	10.4	No notes at this time.	f
214879	Scottish Right	44306	IN	US	Scottish Ale	Year-round	4.0	No notes at this time.	t
320009	MegaMeow Imperial Stout	4378	WA	US	American Imperial Stout	Winter	8.7	Every time this year	f
246438	Peaches-N-Cream	44617	PA	US	American Cream Ale	Rotating	5.1	No notes at this time.	f
8036	World Burp Beer 2002	3469		JP	Japanese Rice Lager	Limited (brewed once)	5.5	No notes at this time.	t

In [11]: `# then grep Stout in the style column
grep("Stout", head(drinks$style))
the word stout in column 1,2 and 4`

1 · 2 · 4

```
In [12]: # then look at te scores of all those beers under abv and
# pull out relevant numbers

drinks$abv[grep("Stout", head(drinks$style))]
```

$7.3 \cdot 10.4 \cdot 8.7$

```
In [13]: mean(drinks$abv[grep("Stout", head(drinks$style))])
# then take the mean, average of the three scores
# since we have achieved for the fisrt six then take
# the head off and run the entire data set
```

8.8

```
In [15]: mean(drinks$abv[grep("Stout", drinks$style)], na.rm=TRUE)
# this is the average score for all the beer.
```

8.20493286813495

```
In [16]: grep("Pale Ale", head(drinks$style))
```

```
In [18]: grep("Pale Ale", head(drinks$style, n=100))
# for the first 100
```

$17 \cdot 18 \cdot 32 \cdot 53 \cdot 81 \cdot 86$

```
In [19]: # to see what they are called
drinks$style[grep("Pale Ale", head(drinks$style, n=100))]
```

'American Pale Ale (APA)' · 'American Pale Ale (APA)' · 'English Pale Ale' ·
 'American Pale Ale (APA)' · 'American Pale Ale (APA)' · 'American Pale Ale (APA)'

```
In [20]: # how many are there?
length(grep("Pale Ale", head(drinks$style, n=100)))
```

6

```
In [21]: # in the whole data set
length(grep("Pale Ale", drinks$style))
```

35689

```
In [22]: grep("Pale", head(drinks$style))
```

```
In [23]: grep("Pale", head(drinks$style, n=100))
```

$15 \cdot 17 \cdot 18 \cdot 23 \cdot 32 \cdot 53 \cdot 81 \cdot 86 \cdot 97$

```
In [24]: drinks$style[grep("Pale", head(drinks$style, n=100))]
```

```
'American Pale Wheat Ale' · 'American Pale Ale (APA)' · 'American Pale Ale (APA)' ·  
'European Pale Lager' · 'English Pale Ale' · 'American Pale Ale (APA)' · 'American Pale Ale (APA)' ·  
'American Pale Ale (APA)' · 'American Pale Wheat Ale'
```

```
In [25]: length(grep("Pale", head(drinks$style, n=100)))
```

```
9
```

```
In [26]: length(grep("Pale", drinks$style))
```

```
45287
```

```
In [27]: grep("Ale", head(drinks$style))
```

```
3 · 5
```

```
In [28]: grep("Ale", head(drinks$style, n=100))
```

```
3 · 5 · 14 · 15 · 17 · 18 · 20 · 28 · 29 · 30 · 32 · 40 · 49 · 50 · 53 · 56 · 62 · 81 · 84 · 86 · 93 · 96 · 97 · 98
```

```
In [29]: drinks$style[grep("Ale", head(drinks$style, n=100))]
```

```
'Scottish Ale' · 'American Cream Ale' · 'American Strong Ale' · 'American Pale Wheat Ale' ·  
'American Pale Ale (APA)' · 'American Pale Ale (APA)' · 'American Black Ale' · 'American Wild Ale' ·  
'American Black Ale' · 'Irish Red Ale' · 'English Pale Ale' · 'American Wild Ale' ·  
'American Amber / Red Ale' · 'Scottish Ale' · 'American Pale Ale (APA)' · 'American Black Ale' ·  
'Belgian Dark Ale' · 'American Pale Ale (APA)' · 'Scotch Ale / Wee Heavy' ·  
'American Pale Ale (APA)' · 'Irish Red Ale' · 'American Brown Ale' · 'American Pale Wheat Ale' ·  
'American Amber / Red Ale'
```

```
In [30]: length(grep("Ale", head(drinks$style, n=100)))
```

```
24
```

```
In [31]: length(grep("Ale", drinks$style))
```

```
107413
```

```
In [32]: grep("Stout", head(drinks$style))
```

```
1 · 2 · 4
```

```
In [33]: grep("Stout", head(drinks$style, n=100))
```

```
1 · 2 · 4 · 11 · 34 · 43 · 44 · 67 · 68 · 73 · 80 · 100
```

```
In [34]: drinks$style[grep("Stout", head(drinks$style, n=100))]
```

'English Oatmeal Stout' · 'Russian Imperial Stout' · 'American Imperial Stout' · 'English Stout' ·
'American Imperial Stout' · 'American Stout' · 'American Imperial Stout' ·
'American Imperial Stout' · 'American Imperial Stout' · 'Russian Imperial Stout' ·
'English Sweet / Milk Stout' · 'English Oatmeal Stout'

```
In [35]: length(grep("Stout", head(drinks$style, n=100)))
```

12

```
In [36]: length(grep("Stout", drinks$style))
```

35943

What is the file size, how many rows, columns and type of data? The file size is 43MB, 358873 rows, 10 columns. Types are integer, character and number

What is the average score for a stout? (consider a stout any named beer from the column name with the word stout in it). It is 8.2

How many Pale Ale's are on this list? (consider a stout any named beer from the column name with the word pale and ale in it) for Pale Ale is 35689, for Pale is 45287, for Ale is 107413 and for Stout is 35943

Question 2

```
In [38]: options(jupyter.rich_display = F)
```

```
In [39]: options(repr.matrix.max.cols=25, repr.matrix.max.rows=25)
```

```
In [40]: head(drinks)
```

	id	name	brewery_id	state	country
1	202522	Olde Cogitator	2199	CA	US
2	82352	Konrads Stout Russian Imperial Stout	18604		NO
3	214879	Scottish Right	44306	IN	US
4	320009	MegaMeow Imperial Stout	4378	WA	US
5	246438	Peaches-N-Cream	44617	PA	US
6	8036	World Burp Beer 2002	3469		JP
	style	availability	abv	notes	
1	English Oatmeal Stout	Rotating	7.3	No notes at this time.	
2	Russian Imperial Stout	Rotating	10.4	No notes at this time.	
3	Scottish Ale	Year-round	4.0	No notes at this time.	
4	American Imperial Stout	Winter	8.7	Every time this year	
5	American Cream Ale	Rotating	5.1	No notes at this time.	
6	Japanese Rice Lager	Limited (brewed once)	5.5	No notes at this time.	
	retired				
1	f				
2	f				
3	t				
4	f				
5	f				
6	t				

```
In [41]: table(drinks$availability)
# check with availability column
```

	Fall	Limited (brewed once)	Rotating
8221		106854	146784
Spring		Summer	Winter
4635		7110	8721
Year-round			
76548			

```
In [43]: length(table(drinks$name[drinks$availability == "Summer"]))
# to know beers available in summer only
# they are so much, so look for Length
```

```
[1] 6349
```

```
In [44]: x <- tapply(drinks$abv[drinks$availability == "Summer"],
drinks$name[drinks$availability == "Summer"],
mean, na.rm = TRUE)

# to first look at the score for them
# split them up according to their name
# then take an average
```

```
In [45]: head(x)
```

""Sugar Daddy"" SPA	7.1
""Summer Dick"" Beaver Dick Brown Pelt Ale	4.8
""The Wind Cried Mari..."" Scottish Heather Ale	4.4
""They Made Me Do It"" Blueberry Ale	4.1
#096 Mashi Mashi No Kuro Summer Stout With Coffee	9.3
#17 Summer Wheat	5.6

```
In [48]: length(x)
```

```
[1] 6349
```

```
In [49]: head(sort(x, decreasing =TRUE))
# most popular with the highest score
```

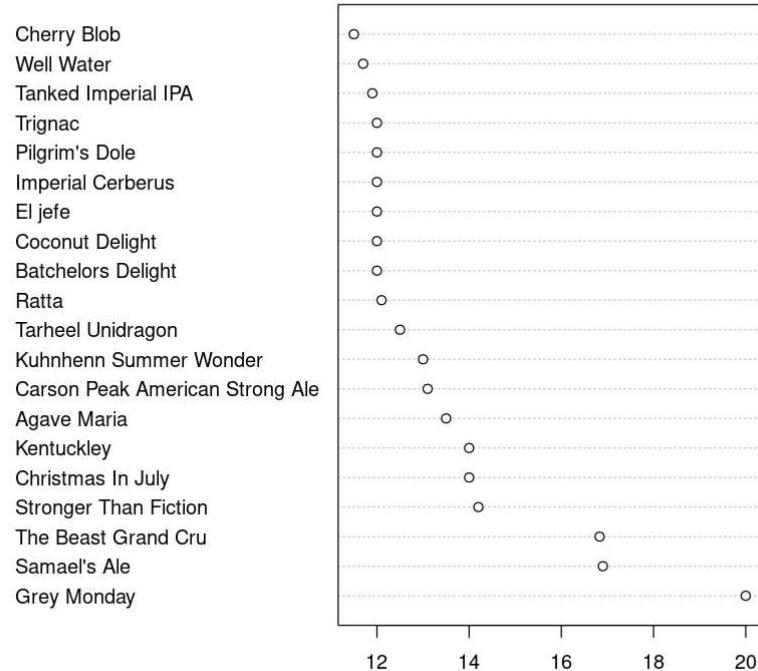
Grey Monday	Samael's Ale	The Beast Grand Cru
20.00	16.90	16.83
Stronger Than Fiction	Christmas In July	Kentuckley
14.20	14.00	14.00

```
In [50]: head(sort(x, decreasing =TRUE), n = 20)
# most top 20 according to their score
```

Grey Monday		Samael's Ale	
	20.00		16.90
The Beast Grand Cru		Stronger Than Fiction	
	16.83		14.20
Christmas In July		Kentuckley	
	14.00		14.00
Agave Maria	Carson Peak American Strong Ale		
	13.50		13.10
Kuhnhenn Summer Wonder		Tarheel Unidragon	
	13.00		12.50
Ratta		Batchelors Delight	
	12.10		12.00
Coconut Delight		El jefe	
	12.00		12.00
Imperial Cerberus		Pilgrim's Dole	
	12.00		12.00
Trignac		Tanked Imperial IPA	
	12.00		11.90
Well Water		Cherry Blob	
	11.70		11.50

```
In [51]: dotchart(head(sort(x, decreasing = TRUE), n = 20))
```

Warning message in dotchart(head(sort(x, decreasing = TRUE), n = 20)):
 "'x' is neither a vector nor a matrix: using as.numeric(x)"



Plot or Graph all the beers that are available in the summer and their ratings

Question 3

```
In [53]: head(drinks)
```

			brewery_id	state	country
1	202522	Olde Cogitator	2199	CA	US
2	82352	Konrads Stout Russian Imperial Stout	18604		NO
3	214879	Scottish Right	44306	IN	US
4	320009	MegaMeow Imperial Stout	4378	WA	US
5	246438	Peaches-N-Cream	44617	PA	US
6	8036	World Burp Beer 2002	3469		JP
	style	availability	abv	notes	
1	English Oatmeal Stout	Rotating	7.3	No notes at this time.	
2	Russian Imperial Stout	Rotating	10.4	No notes at this time.	
3	Scottish Ale	Year-round	4.0	No notes at this time.	
4	American Imperial Stout	Winter	8.7	Every time this year	
5	American Cream Ale	Rotating	5.1	No notes at this time.	
6	Japanese Rice Lager	Limited (brewed once)	5.5	No notes at this time.	
	retired				
1	f				
2	f				
3	t				
4	f				
5	f				
6	t				

```
In [54]: y <- tapply(drinks$abv, drinks$country,
                  mean, na.rm = TRUE)

# now break the data up according to the country
# then Look up the scores for the whole data set
# means Looking at the score, broken up into their mean
```

```
In [55]: length(y)
```

```
[1] 194
```

```
In [56]: head(sort(y, decreasing =TRUE))
```

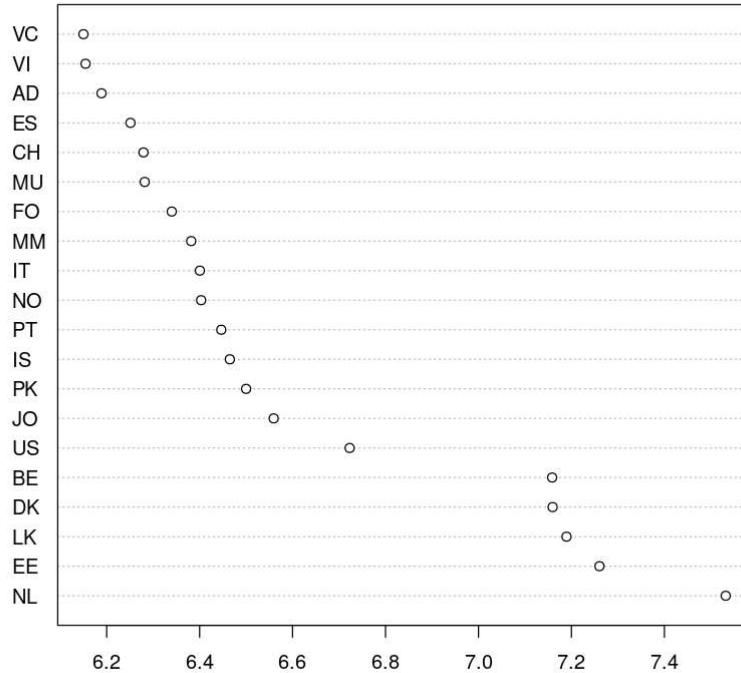
```
NL      EE      LK      DK      BE      US
7.532012 7.260413 7.189474 7.159598 7.158532 6.722863
```

```
In [57]: head(sort(y, decreasing =TRUE), n = 20)
```

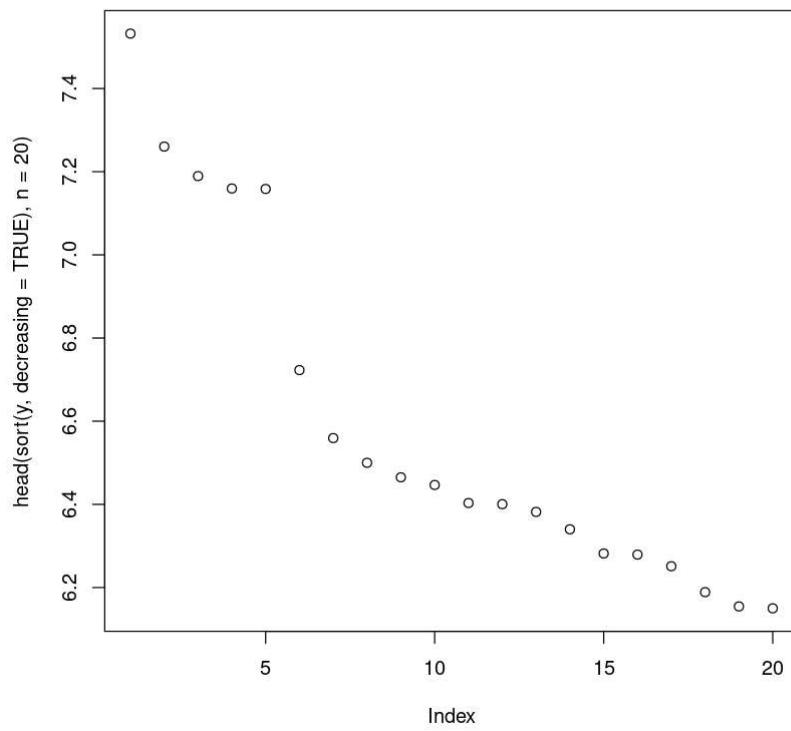
```
NL      EE      LK      DK      BE      US      JO      PK
7.532012 7.260413 7.189474 7.159598 7.158532 6.722863 6.559375 6.500000
IS      PT      NO      IT      MM      FO      MU      CH
6.465049 6.446598 6.403205 6.400460 6.381818 6.340000 6.281818 6.279186
ES      AD      VI      VC
6.251232 6.188889 6.154545 6.150000
```

```
In [58]: dotchart(head(sort(y, decreasing =TRUE), n = 20))
```

Warning message in dotchart(head(sort(y, decreasing = TRUE), n = 20)):
 "'x' is neither a vector nor a matrix: using as.numeric(x)"



```
In [59]: plot(head(sort(y, decreasing =TRUE), n = 20))
```



Create a plot of the average rating of beer by country.

Question 4

```
In [60]: head(drinks$abv[drinks$availability == "Limited (brewed once)"])
# the first six scores for which the
# availability was limited brewed once

[1] 5.5 5.0 11.8 5.2 6.0 6.5
```

```
In [61]: median(drinks$abv[drinks$availability == "Limited (brewed once)"])
# then take their median

[1] NA
```

```
In [62]: median(drinks$abv[drinks$availability == "Limited (brewed once)", na.rm=TRUE])

[1] 6.7
```

```
In [64]: median(drinks$abv, na.rm=TRUE)
# median of the whole data set

[1] 6
```

```
In [65]: table(drinks$availability)

          Fall Limited (brewed once)      Rotating
          8221             106854        146784
          Spring           Summer       Winter
          4635              7110         8721
          Year-round
          76548
```

```
In [66]: unique(drinks$availability)

[1] "Rotating"           "Year-round"
[4] "Limited (brewed once)" "Spring"
[7] "Fall"
```

Do limited runs of beer have a greater median rating than all others? (consider limited to be any beer that has the word Limited in the availability column) Yes

Use the unique function to investigate the availability column. Why are there different labels that are technically the same? From the listed columns the one of rotating is more or less also what could possibly be available in any season of the year and also year round is available all throughout the year, but for pertinent reasons like Fall, Winter and Summer are only available for their specified season while the limited from the name is so limited.

Question 5

```
In [ ]: # code here
```

Markdown notes and sentences and analysis written here.

Pledge

By submitting this work I hereby pledge that this is my own, personal work. I've acknowledged in the designated place at the top of this file all sources that I used to complete said work, including but not limited to: online resources, books, and electronic communications. I've noted all collaboration with fellow students and/or TA's. I did not copy or plagiarize another's work.

As a Boilermaker pursuing academic excellence, I pledge to be honest and true in all that I do. Accountable together – We are Purdue.