STAT240 D100 Spring 2021 SFU

Midterm

This midterm exam consists of 3 problems. All aspects of the midterm exam must be handed in through crowdmark. This midterm exam is open book and take home and due March 5th at 6PM PST. You may access any texts, notes or lectures while completing this midterm exam. You may access resources on the internet provided that you don't communicate any aspect of this midterm exam, distribute the midterm exam material in any way or confer with other students or third parties regarding the midterm exam; as formalized below.

Honour Code

In taking this midterm exam you are required to affirm your willingness to abide with the course policies. By entering your name below, you affirm that you are abiding by the following honour code: I understand that the following activities are prohibited and will be considered cheating. I agree that I will not participate in any of the following activities:

- Looking at or copying from another student's midterm exam or materials while writing the midterm exam.
- Conferring with other students or other parties regarding the midterm exam.
- Having someone else take the midterm exam in your place.
- Distributing the midterm exam materials in any way, or discussing midterm exam materials with anyone in any form or media.
- Misrepresenting the considerations that the midterm exam must be done within the time limitation.

The above honour code is an undertaking for students to abide by both individually and collectively. You must uphold both the spirit and letter of this honour code. Please sign this honour code and upload to crowdmark.

Signature:

Full Name (printed): ANSHAL CHOPRA

Student Number: 301384760

Use a more unique title (reflect your essay content)

Has Data Science Changed the World?

Submitted By: Anshal Chopra

Student Number: 301384760

Submitted To: Dr. Lloyd Elliott

Date: 5th March 2021

SIMON FRASER UNIVERSITY

Has Data Science Changed the World?

Introduction

Data has been important in the recent times and has become the foundation of new inventions and emerging technologies. This importance has led to the increasing amount of data being stored in various formats. Data Science is a very new but hot field in the market that makes use of this data to make decisions that could make a positive change. Now, this change could be either increasing profits for some corporation or deciding what location is the best to make a new railway station. Data has always been used in some form or the other, the field of data science just legitimizes the use of this data by making it more efficient with the help of modern tools and programming languages. Despite all these changes, there still is one question flying around, has data science changed the world? *As data becomes one of the most valuable commodities of all time, data science is a groundbreaking field that makes sure this commodity is not wasted and has an impact on the world*. This essay starts off by discussing the impact of data science in business and industry. It then describes how data is quite a phenomenon for the field of health and medicine and finally concludes the essay using some more examples.

Data science has become a very dominant element in business, industry and science. No wonder, data science is in such huge demand. Steinberg (2020) and Aronovich (2020) in their article, presents a consensus report of the US National Academy of Sciences that shows a consistent and nearly linear increase in web searches regarding data science since 2013, suggesting this as the start of data science as an "intellectual, cultural and economic phenomenon". The ulterior motive of any business is growth, which is only possible by maximizing revenues. They do so my making certain decisions, some of which are intuition driven while other are data driven. Businesses makes use of data to make better informed

decisions. By analyzing previously obtained data, the business can know what to do and what not to do. Caldwell in his blog mentions the scenario of the company Timberland. Timberland was known for making rugged footwear which eventually led it to lose its market share. Due to consumer demand for different products, they were forced to change their line of products and have been progressing ever since. There are other businesses too, which make use of data science. These include streaming services such as Netflix and Amazon to ensure that, people hold onto their services. E-commerce use data so that people can be well informed of the products they are purchasing. But most important of all are social media sites such as Facebook or Twitter which hugely rely on data science for their revenues.

Of course, there is a huge demand for data science in business and industry, but it is not limited to that. Data science has been significant in the medical field as well. Traditionally medicine solely relied on the discretion advised by the doctors, which always was not correct and was prone to human errors, however with advancements in technologies and with the availability of data, it is now possible to obtain accurate diagnostic measures. Liam (2020) talks about several uses of data science in healthcare which includes medical image analysis, drug discovery, genetics and genomics and many more. He adds, drug discovery is an expensive and time-consuming process taking over a period of 12 years and over $2.6 billion since a single formula must pass through a million testing procedures until it gets approved. But with the help of data science, the process is shortened and made much more efficient. Coronavirus or COVID-19 is a hot topic to speak about. Contact tracing is an effective way to slow COVID-19, says Matthews (2020). Data scientists and medical experts are working day and night to make this process more efficient which is, yet another way data science is contributing towards the medical field. Obviously, these are just a few examples of how data science is affecting us in our day to day lives, there might be hundreds of thousands of more such applications.

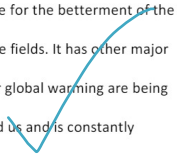
Novel coronavirus

No space

Coronaviruses are a category of viruses. COVID-19 is the name of the disease itself, which is caused by the virus. "SARS-CoV-2" or "novel coronavirus" are names for the specific virus.

Conclusion

In conclusion, no one can deny the importance of data science anymore. Most of the companies today, are making use of data more than ever and the trend seems to follow in the future as well. Use of data in business to increase the revenues and use of data in the field of medicine for the betterment of the people is certainly revolutionary but data science is not just limited to those fields. It has other major significance as well. Many modern-day problems such as climate change or global warming are being worked on with the help of data scientists. Thus, data science has impacted us and is constantly changing the world around us.



Great work!

References

Matthews, K. (2020) *How Data Science Is Being Used to Understand COVID-19*. Retrieved March 4, 2021, from https://www.kdnuggets.com/2020/04/data-science-understand-covid-19.html

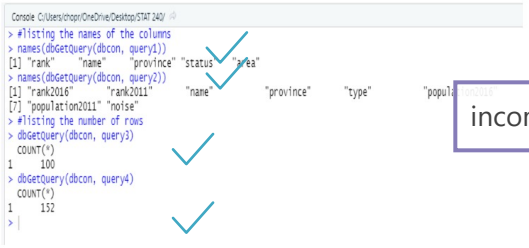
Steinberg DM, Aronovich E. Thoughts on Data Science in business and industry. Appl Stochastic Models Bus Ind. 2020;36:36–40. https://doi.org/10.1002/asmb.2510

Liam, H. (2020) *The Role Of Data Science in Healthcare Advancements: Applications and Benefits.* Retrieved March 4, 2021, from https://datafloq.com/read/role-of-data-science-healthcare-advancements-applications-benefits/8514

Caldwell, S. *5 businesses that benefit from data science.* Retrieved March 4, 2021, from https://www.retaildive.com/ex/mobilecommercedaily/5-businesses-that-benefit-from-data-science

6

```
#####Q2(PART I)#####


#connecting to the database
dbcon = dbConnect(SQLite(), dbname = "stat240.sqlite")

#listing all the tables in the database
dbListTables(dbcon)

#query for selecting all columns from the tables
query1 = "SELECT * FROM citiesA"
query2 = "SELECT * FROM citiesP"

#query for counting the number of rows
query3 = "SELECT COUNT(*) FROM citiesA"
query4 = "SELECT COUNT(*) FROM citiesp"

#listing the names of the columns
names(dbGetQuery(dbcon, query1))
names(dbGetQuery(dbcon, query2))

#listing the number of rows
dbGetQuery(dbcon, query3)
dbGetQuery(dbcon, query4)
```



```
Console C:/Users/chopr/OneDrive/Desktop/STAT 240/
> #listing the names of the columns
> names(dbGetQuery(dbcon, query1))
[1] "rank"     "name"     "province" "status"   "area"
> names(dbGetQuery(dbcon, query2))
[1] "rank2016"       "rank2011"       "name"           "province"       "type"           "popula
[7] "population2011" "noise"
> #listing the number of rows
> dbGetQuery(dbcon, query3)
  COUNT(*)
1      100
> dbGetQuery(dbcon, query4)
  COUNT(*)
1      152
>
```

incomplete answer -2

#####Q2(PART II)#####

#query for selecting unique combinations

```
query5 = "SELECT DISTINCT province, type FROM citiesP
        ORDER BY province,type"
```

#getting unique combinations

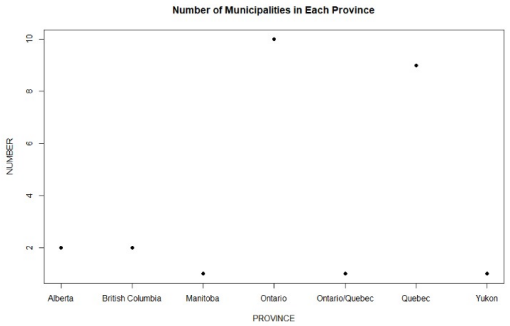
```
dbGetQuery(dbcon, query5)
```

#number of unique combinations

```
nrow(dbGetQuery(dbcon, query5))
```

```
> #getting unique combinations
> dbGetQuery(dbcon, query5)
                    province type
1                    Alberta   CA
2                    Alberta  CMA
3       Alberta/Saskatchewan   CA
4           British Columbia   CA
5           British Columbia  CMA
6                   Manitoba   CA
7                   Manitoba  CMA
8              New Brunswick   CA
9              New Brunswick  CMA
10 Newfoundland and Labrador   CA
11 Newfoundland and Labrador  CMA
12     Northwest Territories   CA
13               Nova Scotia   CA
14               Nova Scotia  CMA
15                   Ontario   CA
16                   Ontario  CMA
17            Ontario/Quebec   CA
18            Ontario/Quebec  CMA
19      Prince Edward Island   CA
20                    Quebec   CA
21                    Quebec  CMA
22              Saskatchewan   CA
23              Saskatchewan  CMA
24                     Yukon   CA
> #number of unique combinations
> nrow(dbGetQuery(dbcon, query5))
[1] 24
>
```

```
#####Q2(PART III)#####


#query for getting common province and cities in citiesP and citiesA
query6 = "SELECT citiesP.name, citiesP.province
        FROM citiesP
        INNER JOIN citiesA ON
          citiesP.name =
          citiesA.name
        ORDER BY citiesP.province, citiesP.name"


#getting common province and cities
Mun_Names = dbGetQuery(dbcon, query6)


#counting the number of municipalities in each province
NumOFMun = Mun_Names %>% group_by(province) %>% summarise(count = n())


#preparing data for the plot
xplot = as.vector(1:nrow(NumOFMun))
yplot = as.vector(NumOFMun$count)


#plotting
plot(xplot, yplot, xlab = "PROVINCE", ylab = "NUMBER", xaxt = "n", main = "Number of Municipalities in
Each Province", pch = 16)
axis(1, at = 1:nrow(NumOFMun), labels = NumOFMun$province, las = 1)
```


Number of Municipalities in Each Province
NUMBER
2
4
6
8
10
Alberta
British Columbia
Manitoba
Ontario
Ontario/Quebec
Quebec
Yukon
PROVINCE

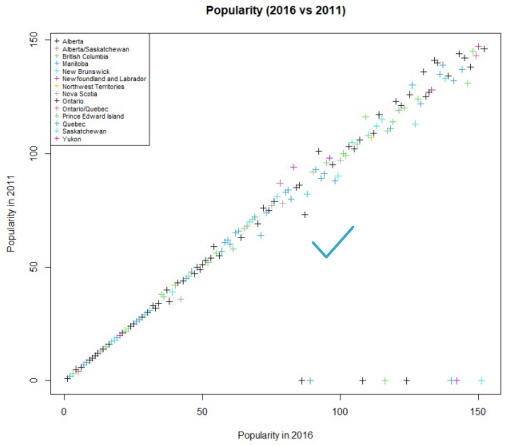
```
#####Q2(PART IV)#####

#query for selecting name, province and ranks from citiesP
query7 = "SELECT name,province, rank2011, rank2016
        FROM citiesP ORDER BY province,name"

#query for getting province names
query8 = "SELECT DISTINCT province FROM citiesP
        ORDER BY province"

#getting query7 and query8
province = dbGetQuery(dbcon, query8)
popular = dbGetQuery(dbcon, query7)

#dealing with values that aren't available
for(i in 1:nrow(popular))
{
 if(popular$rank2011[i] == "NR")
 {
   popular$rank2011[i] = 0
 }
}

#assigning colour(integer) according to the province
colour = c()
for(i in 1:nrow(province))
{
 for(j in 1:nrow(popular))
 {
```

```
  if(popular$province[j] == province$province[i])
  {
    colour[j] = c(i)
  }
 }
}
```

#note: Popularity in 2011 is 0 due to non availability of data

#plotting and labelling

```
plot(popular$rank2016, popular$rank2011, xlab = "Popularity in 2016", ylab = "Popularity in 2011", main = "Popularity (2016 vs 2011)",pch = 3, col = colour)
legend("topleft", legend = province$province, col = c(1:nrow(province)), pch = 3, cex = 0.69)
```



REPORT

Anshal Chopra

4th March 2021

Pitchfork is website that along with various other things, publishes reviews of music albums and rates them out of 10. In the project assigned to me as my midterm for **STAT 240** by **Dr. Lloyd T. Elliott**, I was supposed to scrape out the text of the review and the rating from the Best New Albums section of Pitchfork using R and its various packages. The report emphasizes that the code that I am presenting as my solution works perfectly fine to my knowledge and gives the required output. I will start off by explaining about the packages that I have used. After that I will be focusing on the lines of code in detail and finish off the report by giving two outputs of the code.

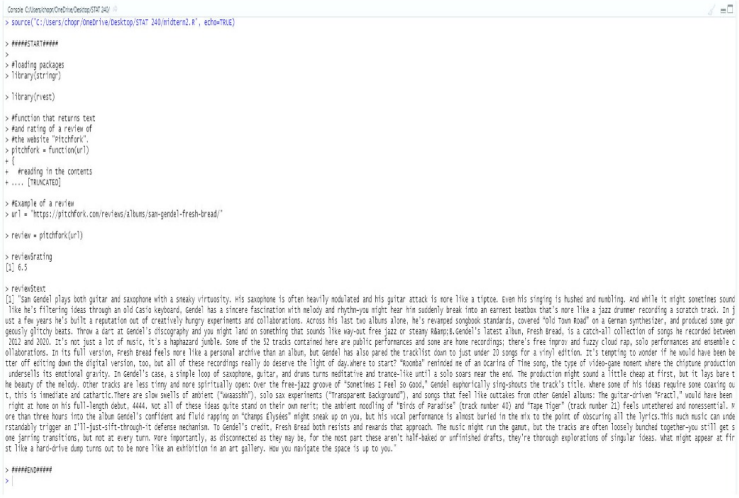
The packages used in the project are **stringr** and **rvest**. The package **stringr** involves a lot of functions that helps in manipulating strings or text. Some of those functions are used in the project. These functions are **substr, str_remove_all** and **str_replace_all** and are explained later in the report. The package **rvest** is relatively a new package and makes it easy to scrape data from html web page. **RCurl** was the package taught in the class, but for some reason it shows an error, so I used **rvest** instead. Functions like **read_html**, **html_text** and **html_nodes** are a part of **rvest**. The required knowledge about the usage of these functions was obtained from several websites including **Datacamp** and **StackOverflow**.

To make a function named **pitchfork** I used **pitchfork = function(url){}** where url is the argument. The function takes in a review link in string format as the argument. Inside the function, the HTML content contained in our webpage is stored in the variable **website**. This was done using the function **read_html**. I went to the site url and clicked on inspect element, which helped me get to know the position of the ratings and the required text. After that, I used **html_nodes** to get the pieces of the HTML code where the required information was. I used **html_nodes** twice, first storing the rating information in the variable **rating_html** and then using it again for **text_html**. The arguments of the function **html_nodes** are the html code data which is **website** in my case and css or xpath nodes to select. The function **html_text** is used to extract only the text information out of the HTML code and that's how I got the rating of the review. After that I used the base r function **as.numeric** to get the rating as a numeric and stored the numeric variable in **rating1**. I also used **as.character** to get the HTML code containing the required text as a string. I didnt use **html_text** again becase I had some unwanted information which was easier to exclude using string manipulation functions rather than using **html_text** function again. I inspected again and got to know I needed information before the <hr> so I located <hr> using **str_locate**. After that, used substr function and got only the part of information i needed by excluding anything that comes after the position of <hr>. Then, i removed all the text contained within <> because it was unwanted and was comprised ,mostly of html code. I did so by using another stringr function which was **str_remove_all** using the regular expression **<.*?>** as an argument. Finally i replaced any more unwanted data using **str_replace_all** and all the required review information was stored in the variablet **text1**. **rating1** and **text1** were later stored in a list as rating and text respectively, so that i could return

the information altogether. After all this had been done, anyone could obtain the information using variable$text or variable$rating.

To show that the code woks perfectly fine i present two examples to you. I will be using the following two url's and extract the ratings and the required text out of it. The links are followed by their respective outputs.

1.) https://pitchfork.com/reviews/albums/sam-gendel-fresh-bread/



```
Console C:/Users/choor/OneDrive/Desktop/STAT 340/
> source("C:/Users/choor/OneDrive/Desktop/STAT 340/midterm2.R", echo=TRUE)

> #####START#####
>
> #loading packages
> library(stringr)

> library(rvest)

> #function that returns text
> #and rating of a review of
> #the website "Pitchfork".
> pitchfork = function(url)
+ {
+   #reading in the contents
+ .... [TRUNCATED]

> #Example of a review
> url = "https://pitchfork.com/reviews/albums/sam-gendel-fresh-bread/"

> review = pitchfork(url)

> review$rating
[1] 6.5

> review$text
[1] "Sam Gendel plays both guitar and saxophone with a sneaky virtuosity. His saxophone is often heavily modulated and his guitar attack is more like a tiptoe. Even his singing is hushed and mumbling. And while it might sometimes sound
 like he's filtering ideas through an old Casio keyboard, Gendel has a sincere fascination with melody and rhythm—you might hear him suddenly break into an earnest beatbox that's more like a jazz drummer recording a scratch track. In j
ust a few years he's built a reputation out of creatively hungry experiments and collaborations. Across his last two albums alone, he's revamped songbook standards, covered 'Old Town Road' on a German synthesizer, and produced some gor
geously glitchy beats. Throw a dart at Gendel's discography and you might land on something that sounds like way-out free jazz or steamy R&amp;B.Gendel's latest album, Fresh Bread, is a catch-all collection of songs he recorded between
 2012 and 2020. It's not just a lot of music, it's a haphazard jumble. Some of the 32 tracks contained here are public performances and some are home recordings; there's free improv and fuzzy cloud rap, solo performances and ensemble c
ollaborations. In its full version, Fresh Bread feels more like a personal archive than an album, but Gendel has also pared the tracklist down to just under 20 songs for a vinyl edition. It's tempting to wonder if he would have been be
tter off editing down the digital version, too, but all of these recordings really do deserve the light of day.Where to start? "Koomba" reminded me of an ocarina of Time song, the type of video-game moment where the chiptune production
 undersells its emotional gravity. In Gendel's case, a single loop of saxophone, guitar, and drums turns meditative and trance-like until a solo soars near the end. The production might sound a little cheap at first, but it lays bare t
he beauty of the melody. Other tracks are less timey and more spiritually open: Over the free-jazz groove of "Sometimes I Feel So Good," Gendel euphorically sing-shouts the track's title. Where some of his ideas require some coaxing ou
t, this is immediate and cathartic.There are slow swells of ambient ("waaaahh"), solo sax experiments ("Transparent Background"), and songs that feel like outtakes from other Gendel albums: The guitar-driven "Fract1." would have been
 right at home on his full-length debut, 4444, Not all of these ideas quite stand on their own merit; the ambient noodling of "Birds of Paradise" (track number 40) and "Tape Tiger" (track number 21) feels untethered and nonessential. M
ore than three hours into the album Gendel's confident and fluid rapping on "Champs Élysées" might sneak up on you, but his vocal performance is almost buried in the mix to the point of obscuring all the lyrics.This much music can unde
rstandably trigger an 'I'll-just-sift-through-it defense mechanism. To Gendel's credit, Fresh Bread both resists and rewards that approach. The music might run the gamut, but the tracks are often loosely bunched together—you still get s
ome jarring transitions, but not at every turn. More importantly, as disconnected as they may be, for the most part these aren't half-baked or unfinished drafts, they're thorough explorations of singular ideas. What might appear at fir
st like a hard-drive dump turns out to be more like an exhibition in an art gallery. How you navigate the space is up to you."

> #####END#####
>
```

2.) https://pitchfork.com/reviews/albums/altin-gun-yol/

```
> #####START#####
>
> #loading packages
> library(stringr)

> library(rvest)

> #function that returns text
> #and rating of a review of
> #the website "Pitchfork".
> pitchfork = function(url)
+ {
+   #reading in the contents
+ .... [TRUNCATED]

> #Example of a review
> url = "https://pitchfork.com/reviews/albums/altin-gun-yol/"

> review = pitchfork(url)

> review$rating
[1] 7.6

> review$text
[1] "Altin Gün's third album arrives with the same mysterious allure as a weirdly shaped parcel found under the Christmas tree. Trapped in lockdown, Amsterdam's finest Turkish psych revivalists started tinkering with drum machines and
 electronics, adding the spacey synth strut of early-'80s disco to their hallucinogenic rock/folk stew. The prospect sounds so charmingly idiosyncratic on paper that you almost dread to press play, for fear that reality will disappoin
t.Remarkably, not only do the results live up to their billing, they also share errant strands of DNA with some surprising strains of contemporary pop. The moody interlocking synth lines of "Ordunun Derelerİ" are only a whisker away fr
om the weeknd's recent excursions into '80s revivalism, and the handclap disco beat and wiggling bassline of "Haçka Yollarİ" is a wayward cousin to Dua Lipa's Future Nostalgia. "Vice Dağ Başında," meanwhile, sits somewhere between Lind
strom's space disco and Gorillaz' occasional excursions into reggae.Damon Albarn's cartoon troupe is a useful reference point for Yol ("road" in English). Much like Gorillaz, Altin Gün concoct their improbable musical fusions in the se
rvice of pop music's joys, rather than eat-your-greens musical worth. Yol represents a fascinating musical laboratory for the way it alloys genres, as the band—founded by Dutch bassist Jasper Verhulst with members of Turkish, Dutch, an
d British descent—traces unusual routes through source material taken largely from the traditional Turkish songbook. But the results are never less than engaging, suggesting the Ventos-in-a-Coke-bottle school of experimentation rather
 than the relentless drudge of high school chemistry. If this year sees a livelier, more jubilant amalgam of disco breaks with saz than Yol's "Hey Karı," then 2021 will have been an outlandishly funky time indeed.Allied to this liberal
 take on genre is Altin Gün's enlightened view of instrumentation, with the band adding synth, congas, drum machines, and even the cosmic twang of the harp-like buzuki ornichord to the psychedelic guitar rock of their two previous albu
ms. This approach might have been inspired by lockdown—the band was stuck at home for three months, swapping demos online—but Yol bubbles with life and adventure, free of the introspective torpor that the pandemic has inspired in some
 artists.Altin Gün also have a wise ear for a tune. The songs on Yol are taken from often archaic sources: "Vekce" is a traditional song from the Anatolian city of Kayseri, while "Arda Boylar" comes from a region of the Balkans that on
ce formed part of the Ottoman Empire. Every song works brilliantly in the neo-disco psych environment; the gorgeous despondency of the folk melodies blends with glittering pop production in a stream of gilded melancholy. It helps that
 Altin Gün are blessed with two extraordinarily gifted singers in Erdinç Ecevit and Merve Dasdemir, whose immaculate technique never comes at the expense of emotive power.Tapping into these antique melodies gives Yol a curious sense of
 timelessness. The album sits somewhere between 19th century Ottoman Empire, '60s Haight-Ashbury, the bubblegum boogie of '90s New York, and the omnivorous pop universe of the present day. To pull off such a trick is clever indeed. But
 it is a mark of Altin Gün's ingenuousness that Yol never feels forced. The album glides along like a particularly elegant swan, musical dexterity and audacious spirit paddling away frantically below the surface."

> #####END#####
>
```

To conclude, from the information provided above including the screenshots of the output of my code, it is pretty evident that the code works perfectly fine and i had clear understanding of it during writing the code. The code i have written is very precise and to the point and each line of that code has some significance and contributes towards the desired output.

```
#####START#####

#loading packages
library(stringr)
library(rvest)

#function that returns text
#and rating of a review of
#the website "Pitchfork".
pitchfork = function(url)
{
 #reading in the contents
 #of the website
 website = read_html(url)

 #extracting the rating in numeric
 rating_html = html_nodes(website,".score" )
 rating1 = html_text(rating_html)
 rating1 = as.numeric(rating1)

 #extracting the review text
 text_html = html_nodes(website,xpath = '//*[@class = "contents dropcap"]')
 text1 = as.character(text_html)

 #removing the non-required content
 position = str_locate(text1, "<hr>")
 text1 = substr(text1, 1, position[1][1]-1)

 #removing content within "< >"
```

```
  text1 = str_remove_all(text1,"\\<.*?>")

  #replacing unwanted characters with new lines.
  text1 = str_replace_all(text1, "\n", "")

  #preparing a list comprising of text and
  #rating for the function output
  list1 = list(text = text1, rating = rating1)

  #returning the list
  return(list1)
}

#Example of a review
url = "https://pitchfork.com/reviews/albums/sam-gendel-fresh-bread/"
review = pitchfork(url)
review$rating
review$text


#####END#####
```