

Books

Shannon Rush

The goal of this analysis is to explore book genre popularity. Some questions I'll try to answer are: Have genre tastes changed over time? Does genre popularity have a pattern in the course of a year? Do demographics such as age or gender tend to report reading the same types of books?

The Data

This data was collected via a public API from a popular book social network. All information is self reported, including demographic information, books read, and genre categorization. Unfortunately it is not necessary when reporting having read a book to categorize it by genre but fortunately many users do.

Two types of data were collected: User data, and information about books users self-reported to be “currently reading”

A note about github: The collected CSV files are too large to share on github. However, you can see exactly how I collected the data at my repo called [‘goodreads-analyses’](#) I’m also happy to share the data I collected if you would like to reproduce my analyses or work with the data yourself. A word of caution: I collected this data too quickly for their tastes and ended up getting IP banned for about a month. Consume at your own risk.

User Data

```
user.cols <- c("userID", "name", "gender", "age", "location", "lastactive",
              "readcount", "currentcount", "wantscount")
users <- read.csv("data/users.csv", header=F, col.names=user.cols, stringsAsFactors=F)
head(users)
```

```
##   userID      name gender age      location lastactive readcount
## 1      1 Otis Chandler  male  36 San Francisco, CA    02/2014       361
## 2      2 odawg Diggity  male  36 San Francisco, CA    11/2013        75
## 3      3      Adrian  male  38 San Francisco, CA    12/2013        52
## 4      4      Isadora female NA San Francisco, CA    06/2013         4
## 5      5 Elizabeth female NA  Santa Monica, CA    01/2014       937
## 6      6      kelly female  40    Oakland, CA    01/2014        28
##   currentcount wantscount
## 1             3         360
## 2             7          19
## 3             0           0
## 4             0           0
## 5             7         568
## 6             1          12
```

```
dim(users)
```

```
## [1] 574919      9
```

Book Data

```
book.cols <- c("userID", "bookID", "shelf", "dateadded", "datestarted", "datefinished", "title",
              "isbn", "isbn13", "imageurl", "pages", "publisher", "publicationdate", "genres")
books <- read.csv("data/currently-reading.csv", header=F, col.names=book.cols,
                 stringsAsFactors=F, na.strings="")
head(books, 1)
```

```
##   userID bookID          shelf          dateadded
## 1      1 123715 currently-reading Fri Nov 08 15:25:13 -0800 2013
##                                     datestarted datefinished
## 1 Fri Dec 27 16:10:42 -0800 2013          <NA>
##                                     title
## 1 Slack: Getting Past Burnout, Busywork, and the Myth of Total Efficiency
##       isbn       isbn13
## 1 0767907698 9780767907699
##                                     imageurl pages
## 1 https://d202m5krfqbp5.cloudfront.net/books/1320419657m/123715.jpg    256
##       publisher publicationdate
## 1 Crown Business          2002
##
## 1 ["business", "management", "non-fiction", "work", "agile", "nonfiction", "software-development", ""]
```

```
dim(books)
```

```
## [1] 130396      14
```

Genres

For each book title collected I also collected the names of the “shelves” it was added to. Many users sort their shelves by genre so this seems to be an adequate way to assign a single common genre to each title with a little processing.

First I’ll find the most common shelf names and decide upon a set of genres to sort all the titles into, if possible.

```
require(stringr)
```

```
## Loading required package: stringr
```

```
genre.counts <- list()
for (genre.set in books$genres) {
  genres <- str_extract_all(genre.set, "[a-z/-]+")[[1]]
  for (genre in genres) {
    if (!genre %in% names(genre.counts)) {
      genre.counts[genre] <- 1
    } else {
      genre.counts[genre] <- genre.counts[[genre]] + 1
    }
  }
}
```

```
sorted.counts <- sort(unlist(genre.counts),decreasing = T)
```

```
sorted.counts[1:100]
```

##	favorites	fiction	non-fiction
##	79091	61488	54406
##	nonfiction	literature	history
##	44533	21635	21470
##	book-club	classics	historical-fiction
##	21357	19401	17115
##	fantasy	novels	biography
##	15497	13007	12454
##	kindle	contemporary	memoir
##	11688	11564	10961
##	philosophy	science	classic
##	10696	9611	9468
##	historical	mystery	politics
##	9137	8578	8489
##	science-fiction	series	romance
##	7622	7536	7527
##	psychology	young-adult	humor
##	7222	6997	6749
##	religion	short-stories	-books
##	6481	6349	6307
##	sci-fi	memoirs	to-buy
##	6066	5773	5597
##	business	self-help	library
##	5436	5133	4935
##	ya	spirituality	essays
##	4763	4163	4061
##	reference	adult-fiction	thriller
##	3923	3819	3781
##	sociology	contemporary-fiction	crime
##	3767	3760	3744
##	travel	favourites	chick-lit
##	3718	3585	3511
##	food	poetry	sci-fi-fantasy
##	3469	3448	3378
##	economics	horror	adventure
##	3284	3275	3236
##	literary-fiction	christian	default
##	3215	3107	3102
##	war	bookclub	art
##	2925	2918	2875
##	american	adult	spiritual
##	2824	2797	2764
##	health	abandoned	magical-realism
##	2756	2694	2520
##	christianity	american-history	dystopia
##	2515	2436	2378
##	audiobook	novel	theology
##	2322	2301	2290
##	paranormal	biographies	childrens

##	2127	2113	2105
##	music	scifi	africa
##	2053	2035	1973
##	ebook	dystopian	wish-list
##	1948	1859	1844
##	suspense	parenting	school
##	1829	1825	1825
##	-	russian	political
##	1821	1791	1766
##	education	france	feminism
##	1654	1647	1622
##	india	french	autobiography
##	1621	1614	1613
##	self-improvement	cooking	magic
##	1598	1559	1543
##	pulitzer	theory	writing
##	1522	1515	1494
##	middle-east		
##	1494		

Now to pick a set of genres that most books will be able to be binned into.

```
genre.bins <- list("history"=c("history","american history","world history","european history","mil. history"),
  "classics"=c("classics","classic"),
  "historical fiction"=c("historical fiction"),
  "fantasy"=c("fantasy"),
  "biography"=c("biography","bio","biographies"),
  "memoir"=c("memoir","autobiography","memoirs","biography memoir","biographies memoir"),
  "philosophy"=c("philosophy"),
  "math and science"=c("science","psychology","sociology","anthropology","economics"),
  "mystery"=c("mystery","mysteries"),
  "politics"=c("politics","political"),
  "science fiction"=c("science fiction","sf","scifi","sci fi"),
  "romance"=c("romance","romances"),
  "young adult"=c("young adult","ya"),
  "humor"=c("humor","comedy","humour"),
  "religion"=c("religion","christianity","spirituality","religions","theology","islam"),
  "business"=c("business","management","marketing","business books"),
  "self improvement"=c("self help","self improvement","professional development","personal development"),
  "reference"=c("reference","art reference","writing reference"),
  "thriller"=c("thriller","thrillers"),
  "poetry"=c("poetry","poet","poetics"),
  "horror"=c("horror"),
  "adventure"=c("adventure"),
  "literary fiction"=c("literary fiction","literary","lit fic","lit fiction"),
  "food"=c("nutrition","foodie","cooking","food","cookbook","cookbooks","recipes"),
  "childrens"=c("childrens","children","kid","kids","children s books"),
  "technology"=c("technology","tech","programming","computer","computers","technical"),
  "comics"=c("comics","comic","graphic novels","graphic novel"),
  "the arts"=c("art","contemporary art","music related","art related","writing","music"))
```

```
require(stringr)
GetGenre <- function(shelf.set) {
```

```

shelves <- str_extract_all(shelf.set, "[a-z/-]+")[[1]]
for (s in shelves) {
  shelf <- str_trim(gsub("-", " ", tolower(s)))
  g <- names(grep(shelf, genre.bins, value=T))
  if (length(g)>0) return(g[1])
}
return(NA)
}

```

```
books$genre <- sapply(books[, "genres"], GetGenre)
```

```
table(books$genre, useNA = "ifany")
```

```
##
##      adventure      biography      business
##           43          2254          1728
##      childrens      classics      comics
##          322          7628          1843
##      fantasy      food historical fiction
##        6424          1310          35902
##      history      horror      humor
##        7143          1251          1267
## literary fiction math and science      memoir
##           38          4911          1322
##      mystery      philosophy      poetry
##        2621          2936          2424
##      politics      reference      religion
##          910          1812          29574
##      romance      science fiction      self improvement
##          972          2556          2461
##      technology      the arts      thriller
##          882          2904          440
##      young adult      <NA>
##        2766          3752

```

```
genre.nas <- subset(books, is.na(genre))
head(genre.nas$genres)
```

```
## [1] "[\"skimmed-incomplete\"]"
## [2] "[\"adult-nonfic\", \"moneysmartweek\"]"
## [3] "[]"
## [4] "[\"haber\"]"
## [5] "[\"blinded-me-with-science\", \"nf-politics-history\"]"
## [6] "[\"ideas-of-the-self\", \"books-on-religion\"]"

```

```
require(stringr)
genre.leftovers <- list()
for (shelf.set in genre.nas$genres) {
  shelves <- str_extract_all(shelf.set, "[a-z/-]+")[[1]]
  for (shelf in shelves) {
    if (!shelf %in% names(genre.leftovers)) {

```

```

        genre.leftovers[shelf] <- 1
      } else {
        genre.leftovers[shelf] <- genre.leftovers[[shelf]] + 1
      }
    }
  }
}

```

Out of the 130396 books only 3752 were unable to be binned. The numbers look pretty reasonable, although 35902 historical fiction did surprise.

I'll create a new CSV so I don't have to re-run this.

```
write.csv(books, file="data/books_with_genre.csv", row.names=F)
```

Obviously most of the interesting analysis is going to come from demographics, so now I'll merge in the user information

```
merged <- merge(books, users)
dim(merged)
```

```
## [1] 130396      23
```

That all looks like it should, so I'll write another csv

```
write.csv(merged, file="data/books_with_users.csv", row.names=F)
```

Analysis

Genres added to currently reading shelf over time

When thinking about genre popularity and the book social network I think the most important date (out of date added to shelf, date started, and date finished) is the date the book was added to the shelf. This is the time when the user is interested in the book.

I'll also include gender, age, and location in this data set. I think these will provide interesting interactive features for the Shiny app.

```
books.users <- read.csv("data/books_with_users.csv")
gt <- subset(books.users, !is.na(genre))
dateadded <- as.Date(gt$dateadded, format="%a %b %d %H:%M:%S %z %Y")
genre.time <- data.frame(dateadded, genre=gt$genre, gender=gt$gender, age=gt$age, location=gt$location)
dim(genre.time)
```

```
## [1] 126644      5
```

```
head(genre.time)
```

```
##   dateadded      genre gender age      location
## 1 2013-11-08    business   male  36 San Francisco, CA
## 2 2012-02-22    religion   male  36 San Francisco, CA
## 3 2012-01-16    religion   male  36 San Francisco, CA
## 4 2010-07-31 science fiction   male  36 San Francisco, CA
## 5 2010-07-30    classics   male  36 San Francisco, CA
## 6 2010-07-15 math and science   male  36 San Francisco, CA
```

```
write.csv(genre.time, file="data/processed_books.csv", row.names=F)
```

Example Plots

Here are some quick plots to see some examples of what we'll be able to do with this data. These are quick and dirty, we'll save the fancy stuff for the shiny app.

```
require(ggplot2)
```

```
## Loading required package: ggplot2
```

```
require(dplyr)
```

```
## Loading required package: dplyr
##
## Attaching package: 'dplyr'
##
## The following objects are masked from 'package:stats':
##
##   filter, lag
##
## The following objects are masked from 'package:base':
##
##   intersect, setdiff, setequal, union
```

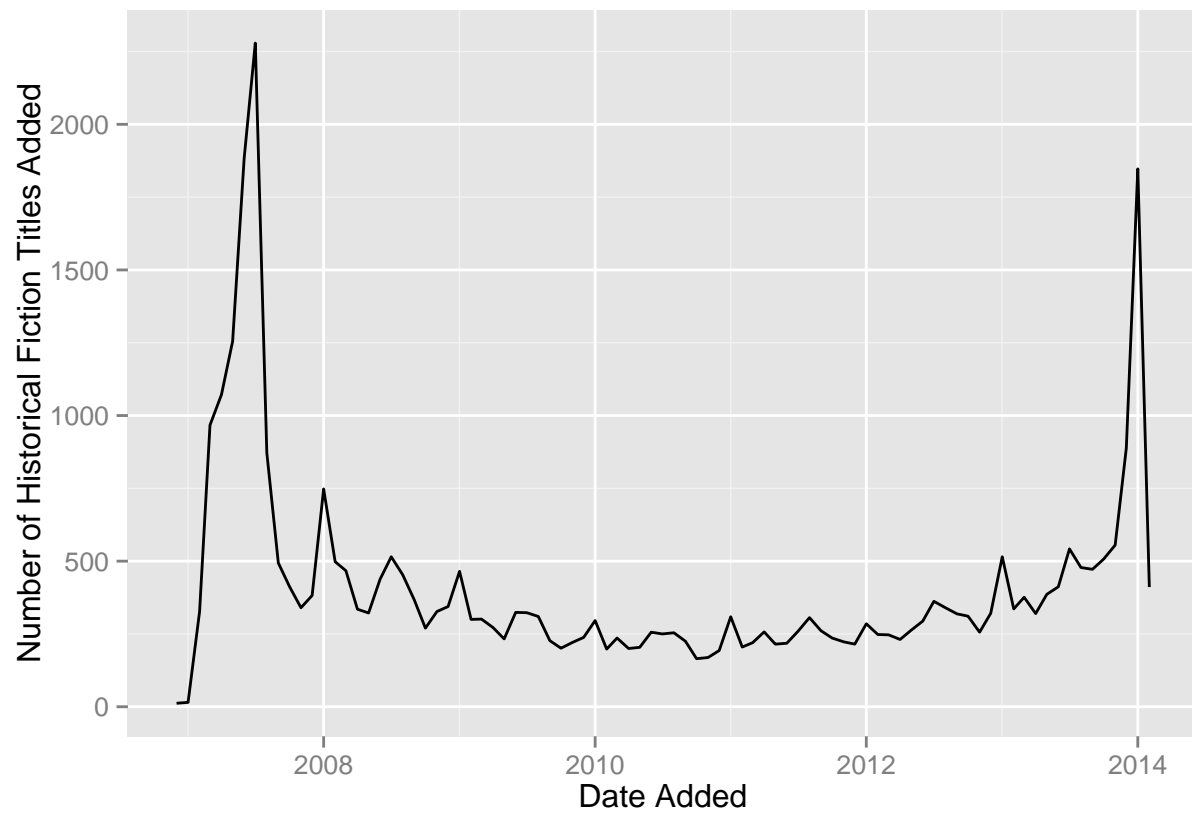
```
require(lubridate)
```

```
## Loading required package: lubridate
```

```
genre.data <- read.csv("data/processed_books.csv")
```

One genre, historical fiction, by date

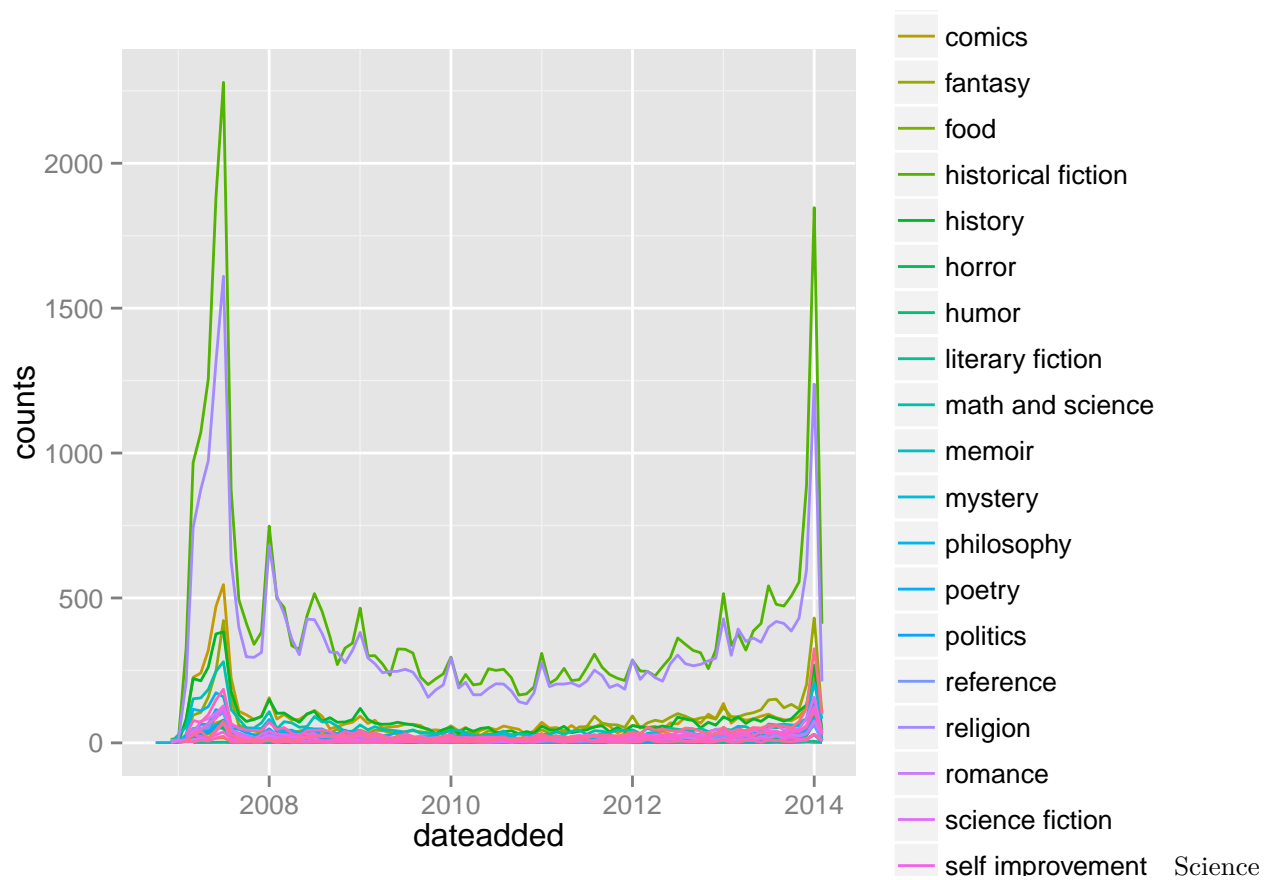
```
hist.fic <- subset(genre.data, genre=="historical fiction")
hist.fic$dateadded <- floor_date(as.Date(hist.fic$dateadded), "month")
by_date <- hist.fic %>%
  group_by(dateadded) %>%
  summarise(counts=n())
print(ggplot(by_date, aes(x=dateadded, y=counts)) + geom_line() + xlab("Date Added") + ylab("Number
```



genres by date

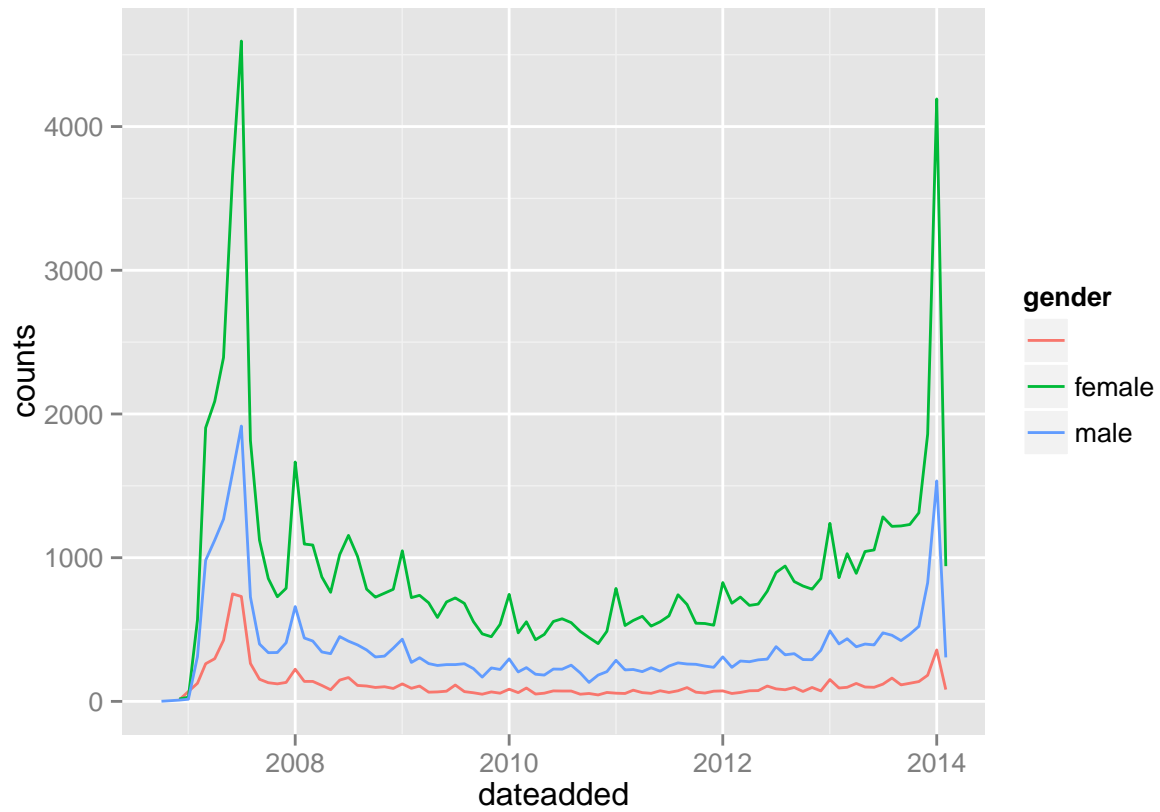
All

```
genre.data$dateadded <- floor_date(as.Date(genre.data$dateadded), "month")
by_genre_and_date <- genre.data %>%
  group_by(genre, dateadded) %>%
  summarise(counts=n())
ggplot(by_genre_and_date, aes(x=dateadded, y=counts, color=genre, group=genre)) + geom_line()
```

Fiction by date and gender

```
sf <- subset(genre.data, genre=="science fiction")
sf_date_gender <- genre.data %>%
  group_by(gender, dateadded) %>%
  summarise(counts=n())
ggplot(sf_date_gender, aes(x=dateadded, y=counts, color=gender, group=gender)) + geom_line()
```



Interactive Shiny App

Now to make it shiny! Visit the code in the ui.R and server.R files in the github repo and visit the shiny app at (https://shanfu.shinyapps.io/shiny_books/)