Scraping our way through the Web

Andrew Tunggal

November 28, 2017

Introduction

A lot of useful information is available on the internet - stores of data are at our fingertips that we can use for our own analytical purposes. It is interesting how much we can obtain for ourselves, and even that we have ways in this day and age to scrape these data from the web. In this post, we will look to perform this procedure on multiple links to showcase the process of obtaining data from the web. Data wrangling can seem to be somewhat of a menial process, but it is very rewarding when you get to the point of actually presenting information.

Necessary Packages

There are some useful packages that help out a lot with the process of web scraping.

```
# Loading the rvest package
library('rvest')
```

```
## Warning: package 'rvest' was built under R version 3.4.2
```

```
## Loading required package: xml2
```

```
## Warning: package 'xml2' was built under R version 3.4.2
```

```
# Loading magrittr package
library('magrittr')
```

```
## Warning: package 'magrittr' was built under R version 3.4.2
```

```
# Loading XML package
library('XML')
```

```
##
## Attaching package: 'XML'
```

```
## The following object is masked from 'package:rvest':
##
##     xml
```

```
# Loading stringr package
library('stringr')
```

Web-Scrapping

For the actual process of webscraping, we go to a website that has data that we desire to use. I chose serenesforest.net, which is a resource for data regarding the game Fire Emblem.

```
#Specifying the url for desired website to be scrapped
grow_url <- 'https://serenesforest.net/path-of-radiance/characters/growth-rates/'
base_url <- "https://serenesforest.net/path-of-radiance/characters/base-stats/"

#Reading the HTML code from the website
webpage_grow <- read_html(grow_url)
webpage_base <- read_html(base_url)
```

So we are able to to read these websites as html or XML files. However, you may get to the link with the data and even be able to read it, but you don't necessarily know how to grab just the table. If you can't grab the table, you won't be able to scrape the part that you want. A useful tool to use to help you is the selector gadget. Using this tool, you can find the aspects of the file that you desire to grab. It will look something like this:

Name	Class	Lv	HP	S/M	Skl	Spd	Lck	Def	Res	Con	Mov	Affin	Weapon ranks
Raven	Mercenary	5	25	8	11	13	2	5	1	8	5		C
Raven HM			29	10	13	15	2	6	2				
Lucius	Monk	3	18	7	6	10	2	1	6	6	5		D
Canas	Shaman	8	21	10	9	8	7	5	8	7	5		B
Dart	Pirate	8	34	12	8	8	3	6	1	10	5		B
Fiora	Pegasus Knight	7	21	8	11	13	6	6	7	5	7		C
Legault	Thief	12	26	8	11	15	10	8	3	9	6		C
Legault HM			29	8	13	17	10	8	4				
Ninian	Dancer	1	14	0	0	12	10	5	4	4	5		–
Isadora	Paladin	1	28	13	12	16	10	8	6	6	8		A, B, D
Heath	Wyvern Rider	7	28	11	8	7	7	10	1	9	7		B
Heath HM			32	13	10	9	7	11	2				
Rath	Nomad	9	27	9	10	11	5	8	2	7	7		B
Hawkeye	Berserker	4	50	18	14	11	13	14	10	16	6		A
Geitz	Warrior	3	40										
Geitz HM			44	19	13	14	10	12	4				B

Looking at this, when you hover over the table, you will find the parts that you want to add to the function “html_nodes” such as below:

```
# obtains the information from the tables on the webpages

# obtains column-names
grow_cols <- webpage_grow %>%
  html_nodes("th") %>%
  html_text()
#obtains table data
grow_stats <- webpage_grow %>%
  html_nodes("td") %>%
  html_text()

# obtains column-names
base_cols <- webpage_base %>%
  html_nodes("th") %>%
  html_text()
#obtains table data
base_stats <- webpage_base %>%
  html_nodes("td") %>%
  html_text()
```

Turning Obtained Information into Useable Data Tables

However, as you may notice, when obtaining this information, we get all the information in one list, which is not necessarily what we want - we want the information sorted into a table. This is may be the case on some websites such as this. Thus, it may require some manual formatting of the data. In the process of scraping data into formats that you desire, sometimes it takes some hard-coding to get what you want. However, when you format it in a organized way, it makes it much easier for your future use.

We will start with the data for growth. If you look at the original link, you will notice the number of columns that there are. We want to as-accurately represent the original table as possible. So we will make vectors containing each corresponding value of that column, and piece it together in our own data frame.

```
# obtaining table for growth rates

g_char_names <- grow_stats[seq(1,length(grow_stats),9)]
g_hp <- grow_stats[seq(2,length(grow_stats),9)]
g_str <- grow_stats[seq(3,length(grow_stats),9)]
g_mag <- grow_stats[seq(4,length(grow_stats),9)]
g_skill <- grow_stats[seq(5,length(grow_stats),9)]
g_spd <- grow_stats[seq(6,length(grow_stats),9)]
g_lck <- grow_stats[seq(7,length(grow_stats),9)]
g_def <- grow_stats[seq(8,length(grow_stats),9)]
g_res <- grow_stats[seq(9,length(grow_stats),9)]

growth <- data.frame("Name" = g_char_names, "HP" = g_hp, "Str" = g_str, "Mag" = g_mag,
                     "Skill" = g_skill, "Spd" = g_spd, "Luck" = g_lck,
                     "Def" = g_def, "Res" = g_res)
```

In the end, it will look like orderly and useable, like this:

```
# shows formatted table for growth rates
growth
```

```
##                 Name     HP    Str    Mag  Skill    Spd   Luck    Def
```

```
##                 Ike     75     50     20     50     55     35     40
##              Titania     80     45     25     60     50     45     40
##                Oscar     55     45     20     50     45     30     35
##                Boyd      75     60      5     50     45     35     25
##                Rhys      40      5     60     50     40     50     25
##              Shinon      75     65     20     70     65     35     50
##              Gatrie      80     55      5     55     25     25     60
##               Soren      45      5     60     55     40     30     15
##                 Mia      50     40     30     45     60     45     20
##              Ilyana      45     25     50     45     30     45     15
##              Marcia      55     40     20     50     55     40     25
##                Mist      50     35     50     25     40     60     15
##                Rolf      60     40     20     45     50     40     30
##               Lethe     130     50      5     65     70     50     40
##            Mordecai     150     65      0     55     50     40     40
##               Volke      65     50      5     55     65     35     20
##              Kieran      60     50     15     50     40     25     40
##                Brom      75     45     10     50     25     20     55
##            Nephenee      55     40     20     55     55     25     35
##              Zihark      55     45     15     50     60     40     30
##               Sothe      60     55     10     70     65     55     35
## 22 Sothe (Blossom *1) 66 2/3 61 1/9 11 1/9 77 7/9 72 2/9 61 1/9 38 8/9
## 23 Sothe (Blossom *2)     84 79 3/4     19     91 87 3/4 79 3/4 57 3/4
## 24                Jill      60     40     30     45     45     25     35
## 25              Astrid      45     40     20     55     50     40     30
## 26             Makalov      60     55      5     45     50     25     45
## 27              Stefan      70     50     20     40     55     25     35
## 28              Tormod      50     20     45     40     45     35     25
## 29              Muarim     145     70      5     70     55     35     60
## 30              Devdan      75     60     30     40     35     40     45
## 31              Reyson      65      5     40     50     50     60     15
## 32                Ulki     140     60     10     65     60     35     35
## 33              Janaff     130     55     10     70     65     40     30
## 34              Tanith      60     40     35     70     40     30     25
## 35              Calill      50     25     45     45     45     30     40
## 36            Tauroneo      60     55      5     50     30     15     60
## 37              Ranulf     130     50      0     55     55     35     35
## 38                Haar      65     60      5     60     35     15     45
## 39              Lucia       70     50     30     70     65     50     40
## 40             Bastian      55     40     65     65     55     30     35
## 41            Geoffrey      65     50     25     55     55     20     45
## 42               Largo      80     70      5     45     45     30     25
## 43             Elincia      60     30     80     45     40     60     25
## 44                 Ena     145     35      5     50     60     40     40
## 45               Nasir     150     50     10     55     45     35     60
## 46              Tibarn     145     70      5     70     65     50     60
## 47             Naesala     135     60     40     70     75     20     55
## 48              Giffca     160     75      5     70     60     40     50
##       Res
## 1      40
## 2      45
## 3      30
## 4      25
## 5      55
## 6      40
## 7      30
## 8      55
## 9      25
## 10     50
## 11     30
## 12     40
## 13     25
## 14     25
## 15     20
## 16     10
## 17     30
## 18     25
## 19     25
## 20     20
## 21     30
## 22 33 1/3
## 23     51
## 24     30
## 25     25
## 26     20
## 27     30
## 28     45
## 29     45
## 30     25
## 31     50
## 32     25
## 33     25
## 34     30
## 35     35
## 36     40
```

```
## 37     20
## 38     20
## 39     40
## 40     50
## 41     45
## 42     20
## 43     35
## 44     30
## 45     25
## 46     25
## 47     35
## 48     30
```

Practice makes Perfect

We will repeat this for our base stats information:

```
# obtaining table for base stats

b_char_names <- base_stats[seq(1,length(base_stats),14)]
b_class <- base_stats[seq(2, length(base_stats), 14)]
b_lvl <- base_stats[seq(3,length(base_stats),14)]
b_hp <- base_stats[seq(4,length(base_stats),14)]
b_str <- base_stats[seq(5,length(base_stats),14)]
b_mag <- base_stats[seq(6,length(base_stats),14)]
b_skill <- base_stats[seq(7,length(base_stats),14)]
b_spd <- base_stats[seq(8,length(base_stats),14)]
b_lck <- base_stats[seq(9,length(base_stats),14)]
b_def <- base_stats[seq(10,length(base_stats),14)]
b_res <- base_stats[seq(11,length(base_stats),14)]

base <- data.frame("Name" = b_char_names, "Class" = b_class, "Level" = b_lvl, "HP" = b_hp,
                   "Str" = b_str, "Mag" = b_mag, "Skill" = b_skill, "Spd" = b_spd,
                   "Luck" = b_lck, "Def" = b_def, "Res" = b_res, stringsAsFactors = FALSE)
```

We have the base stats and growth rates. But something that is an aspect of this game as well is the promotion bonuses that a character receives upon changing classes. Each class has certain stat boosts upon promotion, so these will be taken into account as well. Luckily, this website has this information as well. So we will perform a similar task as was done before for the promotion data:

```
#Specifying the url for desired website to be scrapped
prom_url <- 'https://serenesforest.net/path-of-radiance/classes/promotion-gains/'

#Reading the HTML code from the website
webpage_prom <- read_html(prom_url)

prom_stats <- webpage_prom %>%
  html_nodes("td") %>%
  html_text()

prom_stats <- gsub("+,", "", prom_stats)

# obtaining table for promotion stats

p_class <- prom_stats[seq(1,length(prom_stats),12)]
p_prom <- prom_stats[seq(2, length(prom_stats), 12)]
p_hp <- prom_stats[seq(3,length(prom_stats),12)]
p_str <- prom_stats[seq(4,length(prom_stats),12)]
p_mag <- prom_stats[seq(5,length(prom_stats),12)]
p_skill <- prom_stats[seq(6,length(prom_stats),12)]
p_spd <- prom_stats[seq(7,length(prom_stats),12)]
p_lck <- numeric(length(prom_stats) / 12)
p_def <- prom_stats[seq(8,length(prom_stats),12)]
p_res <- prom_stats[seq(9,length(prom_stats),12)]

prom <- data.frame("Class" = p_class, "Promotion" = p_prom, "HP" = p_hp,
                   "Str" = p_str, "Mag" = p_mag, "Skill" = p_skill, "Spd" = p_spd,
                   "Luck" = p_lck, "Def" = p_def, "Res" = p_res, stringsAsFactors = FALSE)
```

If you look at the resulting table, you will see that there are “+” signs on there. It is fine if you just want to look at the table of information. But if you desire to add up these values, perhaps if you want to simulate final stats, you will want this to be as clean as possible. A way to do this by using the “sapply” function to turn these specific columns into integers:

```
prom[, c(3:10)] <- sapply(prom[, c(3:10)], as.integer)
```

On the note of promotions, you might notice that some characters have access to promotions, while some do not. This may be something that you could desire to take into account in whatever analysis/simulation that you perform on this information. Thus, we will add a column to indicate whether a character has acccess to a promotion or not (and thus, promotion bonuses).

```
# adds logical column indicating whether that character can be promoted or not
base$Prom <- (base$Class %in% prom$Class) | ((paste(base$Class, "(M)")) %in% prom$Class)
```

The table with base stats will now look like this:

```
# shows data table for base stats
base
```

```
##        Name                Class Level HP Str Mag Skill Spd Luck Def Res
## 1       Ike               Ranger     1 19   5   1     6   7    6   5   0
## 2   Titania              Paladin     1 33  12   4    13  14   11  11   7
## 3     Oscar         Lance Knight     3 26   6   1     6   7    5   8   0
## 4      Boyd              Fighter     2 30   7   0     4   6    4   5   0
## 5      Rhys               Priest     4 22   0  10     8   5    8   0  14
## 6    Shinon               Sniper     1 32   9   6    15  13    9   9   6
## 7    Gatrie               Knight     9 31  12   0     6   5    5  14   0
## 8     Soren                 Mage     1 18   0   6     8   8    5   2   7
## 9       Mia             Myrmidon     6 21   7   0    10  13    6   7   2
## 10   Ilyana                 Mage     6 20   1   8    10   9    6   3  10
## 11   Marcia       Pegasus Knight     5 20   8   0     7  11    4   8   6
## 12     Mist               Cleric     1 16   1   4     4   7    6   2   7
## 13     Rolf               Archer     1 18   5   0     8   6    4   6   2
## 14    Lethe    Beast tribe (Cat)     3 34  12   4    10  12   15   9   7
## 15 Mordecai  Beast tribe (Tiger)     2 41  15   2     8   8   10  13   4
## 16    Volke                Thief    10 25  12   0    13  13    7   7   3
## 17   Kieran           Axe Knight    12 30  11   1    10  12    8  10   1
## 18     Brom               Knight     8 28  10   1     9   7    4  13   2
## 19 Nephenee              Soldier     7 22   8   2    10  11    6   9   3
## 20   Zihark             Myrmidon    10 25  10   1    13  15    6   7   0
## 21     Jill         Wyvern Rider     8 24  11   0    10   9    6  11   2
## 22    Sothe                Thief     1 20   5   1     7  11    5   4   0
## 23   Astrid           Bow Knight     1 20   6   2     6   7    3   5   4
## 24  Makalov         Sword Knight    10 30   9   2     7  10    8  10   2
## 25   Stefan          Swordmaster     8 38  19   8    27  25    5  12   9
## 26   Muarim  Beast tribe (Tiger)     9 45  16   4    13  15   11  12   5
## 27   Tormod                 Mage     7 20   2  10     9   9    8   4   9
## 28   Devdan           Halberdier     4 36  14   7    15  13   16  11  10
## 29   Tanith        Falcon Knight    10 32  16  10    18  24   18  15  13
## 30   Reyson   Bird tribe (Heron)     3 22   1  10    11  14   15   2  20
## 31   Janaff    Bird tribe (Hawk)     8 39  13   5    15  17   16  11  10
## 32     Ulki    Bird tribe (Hawk)     7 41  15   4    14  12   10  14   9
## 33   Calill                 Sage     6 32   8  19    18  18   16   8  17
## 34 Tauroneo              General    14 48  22  11    18  13   14  22  14
## 35   Ranulf    Beast tribe (Cat)     9 46  19   4    17  17   13  17   6
## 36     Haar          Wyvern Lord    11 47  21   8    19  17   12  20  10
## 37    Lucia          Swordmaster    12 36  15  12    21  23   16  10   8
## 38  Bastian                 Sage    13 35  12  19    21  16   15  12  20
## 39 Geoffrey              Paladin    11 43  18   9    17  19   12  21   9
## 40    Largo            Berserker     7 52  21   4    21  20   12  10   3
## 41  Elincia      Princess Crimea     1 27   9  12    16  18   15  11  15
## 42      Ena   Dragon tribe (Red)    10 52  20   9    17  15   14  23  21
## 43    Nasir Dragon tribe (White)    18 56  20  11    23  22   17  24  27
## 44   Tibarn    Bird tribe (Hawk)    18 63  30  11    31  24   24  26  19
## 45  Naesala   Bird tribe (Raven)    17 57  25  15    26  31   19  21  16
## 46   Giffca   Beast tribe (Lion)    20 68  32  10    28  25   22  25  16
## 47 Sephiran               Bishop    10 42   4  29    22  14   30  12  30
## 48   Leanne   Bird tribe (Heron)     1 20   0  12    13  13    7   1  23
##     Prom
## 1   TRUE
## 2  FALSE
## 3   TRUE
## 4   TRUE
## 5   TRUE
## 6  FALSE
## 7   TRUE
## 8   TRUE
## 9   TRUE
## 10  TRUE
## 11  TRUE
## 12  TRUE
## 13  TRUE
## 14 FALSE
## 15 FALSE
## 16  TRUE
## 17  TRUE
## 18  TRUE
## 19 FALSE
## 20  TRUE
## 21 FALSE
## 22  TRUE
## 23 FALSE
## 24  TRUE
## 25 FALSE
## 26 FALSE
## 27  TRUE
## 28 FALSE
## 29 FALSE
## 30 FALSE
## 31 FALSE
```

```
## 32 FALSE
## 33 FALSE
## 34 FALSE
## 35 FALSE
## 36 FALSE
## 37 FALSE
## 38 FALSE
## 39 FALSE
## 40 FALSE
## 41 FALSE
## 42 FALSE
## 43 FALSE
## 44 FALSE
## 45 FALSE
## 46 FALSE
## 47 FALSE
## 48 FALSE
```

Something that you may notice is that most RPG characters have maximum stats. This means that no matter the growth rates of a character, you cannot exceed those maximum stats. This website also provides information on the maximum stats of each class of units. We will perform this process again. Hopefully you've gotten the hang of this, so it shouldn't be too difficult for you:

```
#Specifying the url for desired website to be scrapped
max_url <- 'https://serenesforest.net/path-of-radiance/classes/maximum-stats/'

#Reading the HTML code from the website
webpage_max <- read_html(max_url)

max_stats <- webpage_max %>%
  html_nodes("td") %>%
  html_text()

# obtaining data table for maximum stats
m_class <- max_stats[seq(1,length(max_stats),9)]
m_hp <- max_stats[seq(2,length(max_stats),9)]
m_str <- max_stats[seq(3,length(max_stats),9)]
m_mag <- max_stats[seq(4,length(max_stats),9)]
m_skill <- max_stats[seq(5,length(max_stats),9)]
m_spd <- max_stats[seq(6,length(max_stats),9)]
m_lck <- max_stats[seq(7,length(max_stats),9)]
m_def <- max_stats[seq(8,length(max_stats),9)]
m_res <- max_stats[seq(9,length(max_stats),9)]

max <- data.frame("Class" = m_class, "HP" = m_hp, "Str" = m_str, "Mag" = m_mag,
                  "Skill" = m_skill, "Spd" = m_spd, "Luck" = m_lck,
                  "Def" = m_def, "Res" = m_res, stringsAsFactors = FALSE)
```

There is an asterisk on some characters, as you may notice. As indicated on the website, this indicates that these are unused in the game. This means that, for our interest, we can remove these rows:

```
# removes asterisked-rows from data table
max <- max[-c(24, 34, 38, 46, 53), ]
max
```

```
##                    Class HP Str Mag Skill Spd Luck Def Res
## 1   Non-promoted physical 40  20  15    20  20   40  20  20
## 2             Lord / Hero 60  26  20    27  28   40  24  22
## 3          Swordmaster (M) 60  24  20    29  30   40  24  22
## 4          Swordmaster (F) 60  22  20    29  30   40  22  25
## 5               Halberdier 60  25  20    28  26   40  28  25
## 6                  Warrior 60  30  20    28  27   40  25  20
## 7                   Sniper 60  25  20    30  28   40  25  23
## 8                  General 60  29  20    27  24   40  30  25
## 9  General (Black Knight) 70  30  20    30  35   40  30  30
## 10       Horse Knight (F) 40  20  15    20  20   40  20  15
## 11             Paladin (M) 60  26  20    26  27   40  27  25
## 12             Paladin (F) 60  25  20    26  27   40  27  26
## 13           Falcon Knight 60  23  20    26  28   40  24  27
## 14         Princess Crimea 60  20  25    26  28   40  24  27
## 15         Wyvern Lord (M) 60  29  20    28  26   40  29  22
## 16         Wyvern Lord (F) 60  27  20    26  27   40  27  25
## 17              King Daein 80  40  40    40  40   40  40  40
## 18                    Mage 40  10  20    20  20   40  10  20
## 19                    Sage 60  15  30    28  28   40  20  28
## 20         Priest / Cleric 40  15  20    20  20   40  20  20
## 21                  Bishop 60  15  29    22  25   40  20  30
## 22                Valkyrie 60  20  26    24  26   40  20  29
## 23            Assassin (M) 60  23  20    30  30   40  22  20
## 25               Berserker 60  30  20    24  28   40  26  20
## 26     Beast tribe (Lion) 80  32  20    35  33   40  35  27
## 27    Beast tribe (Tiger) 75  30  20    33  34   40  30  24
## 28    Beast tribe (Cat M) 70  29  20    34  35   40  30  24
## 29    Beast tribe (Cat F) 70  26  20    34  36   40  27  27
## 30                   Lion 80  40  20    39  36   40  40  30
## 31                  Tiger 75  37  20    37  37   40  35  27
## 32                Cat (M) 70  35  20    38  38   40  35  27
## 33                Cat (F) 70  32  20    38  39   40  32  30
## 35   Dragon tribe (White) 80  30  25    30  32   40  35  35
## 36   Dragon tribe (Red M) 80  35  20    31  32   40  36  30
## 37   Dragon tribe (Red F) 75  35  20    31  31   40  36  30
## 39           White Dragon 80  40  25    36  35   40  40  40
## 40         Red Dragon (M) 80  45  20    35  35   40  40  35
## 41         Red Dragon (F) 75  40  20    35  35   40  40  35
## 42      Bird tribe (Hawk) 65  26  20    35  36   40  26  26
## 43    Bird tribe (Tibarn) 75  33  20    35  36   40  32  29
## 44     Bird tribe (Raven) 65  25  20    31  34   40  25  31
## 45   Bird tribe (Naesala) 70  29  24    34  37   40  27  32
## 47   Bird tribe (Heron M) 60  10  20    17  26   40  15  35
## 48   Bird tribe (Heron F) 60  10  20    17  25   40  15  35
## 49                   Hawk 65  32  20    40  39   40  30  30
## 50          Hawk (Tibarn) 75  40  20    40  39   40  35  30
## 51                 Raven 65  30  21    35  38   40  28  35
## 52        Raven (Naesala) 70  35  25    38  40   40  30  35
## 54          Heron (White) 60  10  25    20  30   40  16  40
## 55       Civilian / Child 60  20  20    20  20   40  20  20
```

Saving wrangled Data

So now we have tables of the data for the growth rates, the base stats, the promotion bonuses, and the maximum stats for each character and class. Save each as a .csv file:

```
# creates .csv files for the data tables that we have made
write.csv(prom, file = "fe9_promotions.csv")
write.csv(base, file = "fe9_base.csv")
write.csv(growth, file = "fe9_growth.csv")
write.csv(max, file = "fe9_max.csv")
```

Conclusion

With these data, you have different possibilities that you can do with them. Web scraping is an important skill to have if you want to be able to use data in your work. The more effort you put into wrangling your data, the easier it makes the computational aspect of your work/research.

Resources

http://selectorgadget.com/

https://serenesforest.net/path-of-radiance/characters/base-stats/

https://serenesforest.net/path-of-radiance/characters/growth-rates/

https://serenesforest.net/path-of-radiance/classes/maximum-stats/

https://serenesforest.net/path-of-radiance/classes/promotion-gains/

https://stackoverflow.com/questions/2288485/how-to-convert-a-data-frame-column-to-numeric-type

https://stackoverflow.com/questions/2667673/select-first-4-rows-of-a-data-frame-in-r

https://stackoverflow.com/questions/13774773/check-whether-value-exist-in-one-data-frame-or-not