Scraping process

Tales of Phantasia: Narikiri Dungeon X was used as an example game to see how the scraping process would be. From its game page, I learned that the most interesting variables would be:

Game title - obvious.

Number of owners - number of GameFAQs users who claim to own the game.

Rating - average rating, which is determined by GameFAQs users. The scale is out of 5.

Numbers of owners rated - number of GameFAQs users who claim to own the game and also gave said game a rating.

Difficulty - the difficulty of the game which is determined by a popular vote. The difficulty ratings are "Easy", "Easy-Just Right", "Just Right", "Just Right-Tough", "Tough", "Tough-Unforgiving" and "Unforgiving".

Percentage of difficulty ratings - obvious.

Game length - average game length, which is determined by GameFAQs users. Unit is in hours.

Votes for game length - number of GameFAQs users who gave how long it took for them to beat the game.

Number of users who completed the game - obvious.

Completion rate - determined by dividing the number of users who completed the game by the number of owners.

After determining my variables, I used the R libraries "rvest", "magrittr" and "XML" and Mozilla Firefox's extract operation to scrape Tales of Phantasia: Narikiri Dungeon X's page. Because this process would be used on hundreds, if not thousands, of other games as well, I made a function that generalized the process.

The trickier part was obtaining the urls of all the games that I would scrape from. At the time of the presentation proposal, I was only able to extract data from all the PSP RPG's. Analyzing the url "https://gamefaqs.gamespot.com/psp/category/48-role-playing?page=1", I noticed that:

-"psp" can be replaced with any system's acronym

-"?page=1" is used to change pages (first page ends with "?page=0")

-not really needed for the PSP in particular, but for newer systems, we need to add "&dist=1" at the end of the url to only consider retail games. We don't want to accidentally add downloadable content since they are almost always not full games.

I used the inspect function to determine where the locations of the links to first PSP RPG and last PSP RPG on the page. I learned that the location of the link to the first RPG on the page varies(which is also the main reason why I never made a function for obtaining all the RPG urls for any game system) and that the location of the link to the last RPG on the page is always 35 indices before the last element.

I was able to get all the PSP RPG urls by noticing that each game url has exactly 2 /'s and always contain "/psp/". After getting all the PSP RPG urls, I used my scraping function to extract all their relevant game data.

Light exploratory data analysis

The last thing that I wanted to do was make sure that my variables actually make sense, based on my experience with playing PSP RPG's. Game length having a wide range made sense because RPG's can range from extremely short to extremely long depending on how much content it has. Completion range also having a wide range makes sense because RPG's do require commitment since they can't be beaten in one sitting. It was also interesting, but not that surprising, to see that as the completion rate goes up, the rating of the game also tended to increase. It's usually hard for players to want to beat games that they're not enjoying. Lastly, RPG's are usually not that tough but also not that simple so "Just Right" being the most popular difficulty made sense as well.