Problem Set 4 - ECON 5253

Thomas Mondry

February 20, 2022

1 Data sources I'd like to scrape from

- Niche example – the J! Archive is a fan-created site which maintains full clue-by-clue data about every televised game of *Jeopardy!*. In a data science class last semester which concentrated mostly on modeling techniques, my group (which included Paul) attempted to predict whether or not a contestant would become a great player (i.e. win more than 3(?) games) using any available information from the first game. We used data we found on the internet which had been scraped from the J! Archive, which does not have an API. There were a lot of issues with the data & I'm considering doing something with *Jeopardy!* for my project in this class as well, with some different "research question" in mind, but if I did so I would scrape the data myself to try to fix some of the problems we encountered last time.
- Another topic I'm considering for my final project is something related to tennis; I could scrape data from the Tennis Abstract site, which is also a fan-created site with detailed point-by-point information from thousands of tennis matches. A few people have written about why, given the money & international scope of tennis, it has been so slow to embrace analytics like most other major sports have. Part of the problem is that much of the strategy of the game is highly dependent on the idiosyncratic qualities of both a specific player and their specific opponent. As data becomes more widely available, I'm interested in seeing if a dimension reduction or clustering process can be developed so that any player's style can be reliably reduced to a few quantifiable attributes; if this type of approach ends up being feasible, it could be used to develop a general model of tennis strategy that takes both players' attributes as parameters.

2 Some answers from previous problems

Problem 5 (JSON practice): `mydf` is a tibble; `mydf$date` is a character vector, or a character tibble if obtained through the tidyverse.

Problem 6 (sparklyR practice): `df1` is a normal tibble, while `df` appears to be a tibble with some different properties for use with Spark and SQL. The periods in the column names of `df1` are replaced with underscores in those of `df`. This is likely because Spark can't work with periods in object names; SQL certainly can't.