Ben Kopchains

Prof. Barr

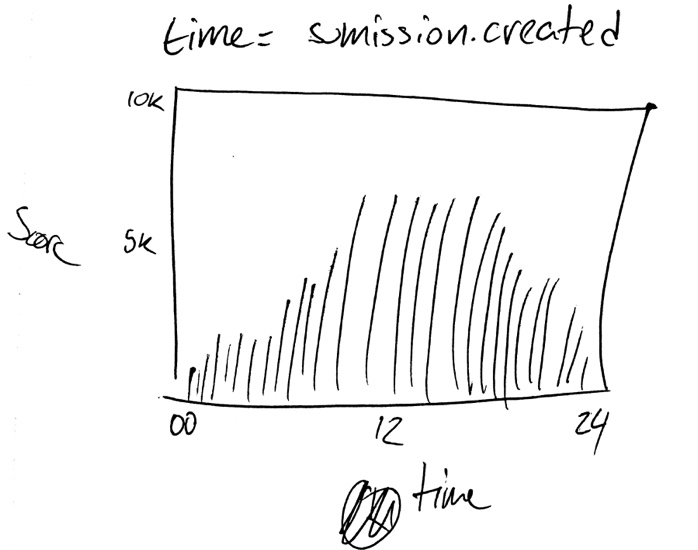
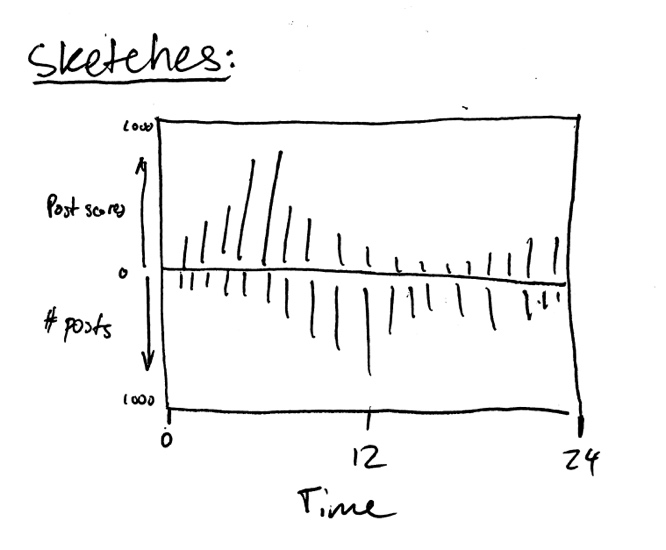
CSC-234

13 March 2016

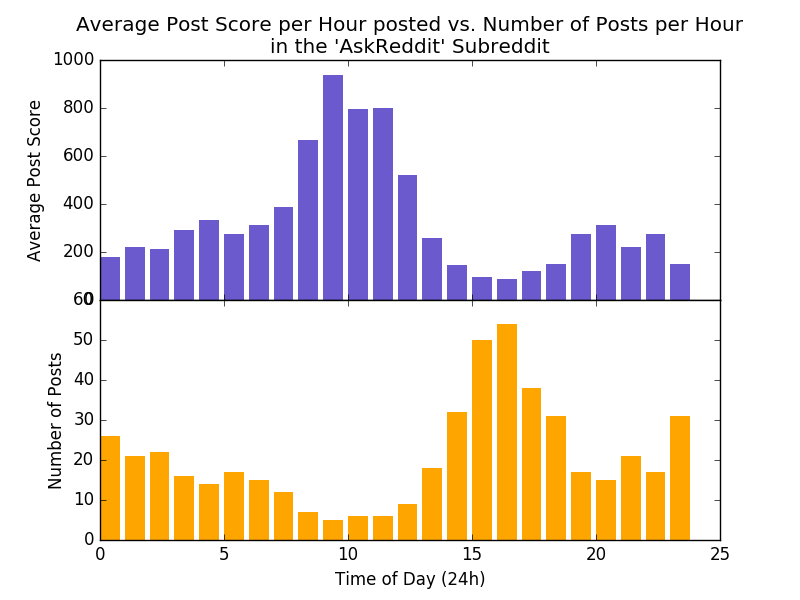
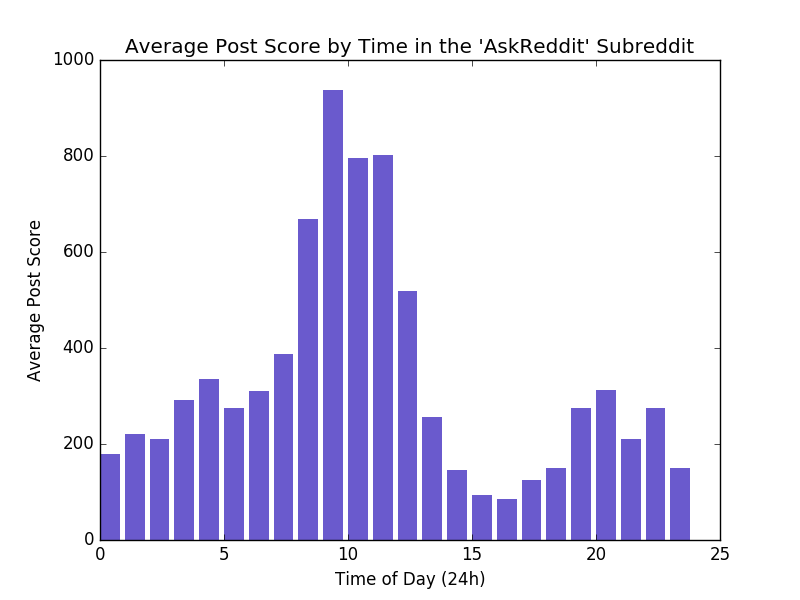
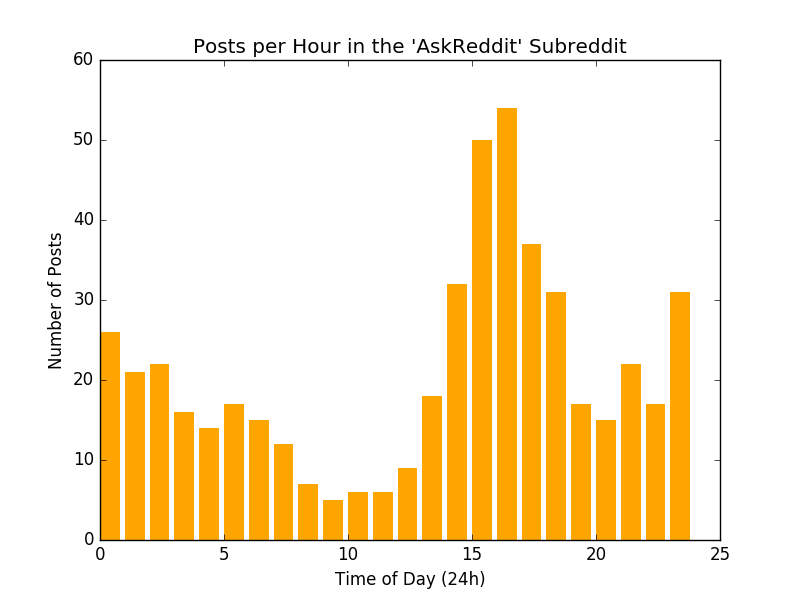
Final Project: Part III

In the third portion of the final project for this class, the students were given free choice for the topic and data set that they planned to analyze. After some research, I decided to explore data related to the popular link-sharing social website, Reddit. Reddit allows users to not only share links and text that they find interesting, but also allows others to “upvote”, “downvote”, and comment on each post. Each post is then moved up and down on the front page of the site, based on its overall score (upvotes vs downvotes). I decided to ask the question “when should I post to reddit?” in order to determine when and where a user should post in order to create a successful post. From there, I asked: “what makes a successful post?”, discovering that the most relevant factors were post time, post location, and the post’s score.

In order to collect the relevant data, I was tasked with using Python to scrape Reddit’s HTML code in order to find each post’s attributes. Unfortunately, Reddit’s code does not allow for users to simply use python’s basic libraries to access the website, but they do provide their own library to the public, for users who still wish to scan the site. Reddit’s python library, PRAW, allowed me to easily iterate through a multitude of posts and record their scores, post times, and other data. After more research, I discovered that one of the two most interesting sets of data would be comparisons between initial post time, and final post score. The other data set I wanted to look into involved a certain type of post: “ask\_\_\_” posts. This post type will generally be in a section or “subreddit” of Reddit related to question and answer comment threads. The subreddit that I analyzed for the majority of this data was AskReddit.



The data I aimed to analyze in the second part of my exploration was related to the first, but involved the types of questions each post asked. Looking at who, what, where, when, why, and how, I also wanted to scrape Reddit to find how successful each post was, based on the type of question asked, effectively revealing how willing people are to answer a question is simply based on the type. My initial sketches were very basic, and related to the first set of data (general success) more than the second (question types). In the first sketch (left) I ended up inadvertently predicting how the first data set would look, and the second (right) was also close, but the labels were not the same as the final output. The visualizations I intended to create were bar charts, with three graphs for each subreddit: score vs. time, number of posts vs. time, and a comparison of the results of the previous two. I believed that my questions were answerable from this type of visualization because I would easily be able to see the trend as the number of posts & successful posts rise and fall on each hour of the day.



The visualizations I ultimately built were very similar to my sketches: with two separate graphs detailing posts per hour and average post scores by time, and a third graph of the two, sharing an x-axis in order to be able to more easily compare the two. What I found interesting was hard to see, looking at the first two separately, but when looking closely at the third comparison graph, it becomes clear that posts per hour, and average post score per hour are – for the most part – inversely proportional. For me, this information was surprising: I was expecting to just see similar trends in both graphs, with a significant drop while users are at work or asleep, and peaks while users are just getting home, or up late at night.

Overall, the task of researching and creating these sets of visualizations was very much an exploratory experience. Going into the programming and coding, I wasn’t sure what to expect the outcome of the graphs to be, and upon the discovery that the trends between the two data types (score per hour and posts per hour) were almost always inversely proportional, I was very pleasantly surprised. Utilizing Reddit’s PRAW library, my python code asks a user for a sample size and a specific subreddit. The code is designed to be modular, in that it can take in any valid subreddit, and be used by anyone to explore the sections of Reddit of their choosing. In closing, I believe that this project was very interesting, and revealed to me a good amount of data I had never even thought about before.