Proposal for Runner Density

We have been trying to implement a function that displays the number of runners at a given segment of arbitrary distance in the race. Previously, we were using a function that gave us a step function. After, we used a function that implemented some smoothing of the step function. None of those were real depictions of how the bell-curve (the real runners) move throughout the race. Thus, I propose a simulation.

I scraped data for every individual runner from the 2008 through 2014 marathons. I identified the % of runners that are placed in each corral based on average times for previous marathons and corral assignment given times for the 2015 marathon. I also used all the scraped data to obtain a function that determines a runner's speed based on temperature and corral. The data shows a clear difference in runner's speed based on:

- Average temperature on day of the marathon
- Distance travelled so far
- Corral

*Note: I do not have detailed temperature data (hourly or ½ hour measurements) to know how runner's speed change with real changes in temperature, but this would improve the simulation

With this data, I simulated a marathon with n runners and m minutes.

- Generate n runners
- Assign a % of runners to each corral
- Assign a speed function to each runner based on:
    - Avg corral speed
    - Temperature
    - Distance traveled
    - Standard deviation of speed for that corral
- Assign start times based on real start intervals
    - Corrals begin at designated time based on the 2015 information
    - Runners within a corral start the race 'uniformly' (assumption)
- Each minute of the race
    - Calculate speed of each runner at minute t based off position from previous minute $t-1$
    - "Move" the runner 1 minute at calculated speed
    - If position>42.2, runner finished

Then I calculate density at a given minute t

- Calculate positions for all runners at time t
- "Group" them by segment (segment = arbitrary measurement of 1 mile, 1 km, 5km, etc)
- Count them
- Count number of people who finished

RESULTS

Checking with real information from the 2014 marathon, given the 2014 marathon temperatures, there were very few discrepancies in the measurements. Most of the time, the simulation was within 5% of the 10-minute 5k segment real data. The only times when the simulation was off (20-25%) was when a large group of runners was at an extreme point of one segment *(e.g. the simulation puts them at segment 2 when in reality they just transitioned to segment 3)*

IMPROVEMENTS

We previously thought about using the real-time feed of 5k cumulative measurement as a sort of feedback to our simulation. This poses a problem to our simulation because <u>the data we get is de-identified from any runner, corral, speed, etc.</u> and also because <u>we get data every 5k rather than every mile</u>. I brainstormed some methods to 'modifying' the simulation (changing the speed or positions of some runners) and initial tests gave a very small improvement.

A drawback of attempting to change the simulation is that we have to re-calculate positions for 40,000 runners, which is time-intensive, cumbersome, and from quick programming attempts, not very useful.

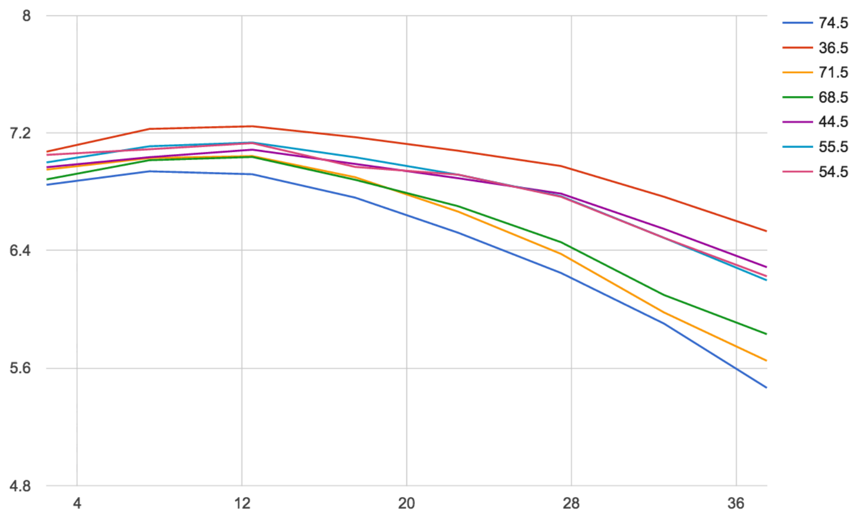
PROPOSAL

Since we cannot 'match' simulated runners to real runners or efficiently modify their speeds, I propose that we simulate the marathon given a temperature and number of runners. We can easily display this simulation on the map.

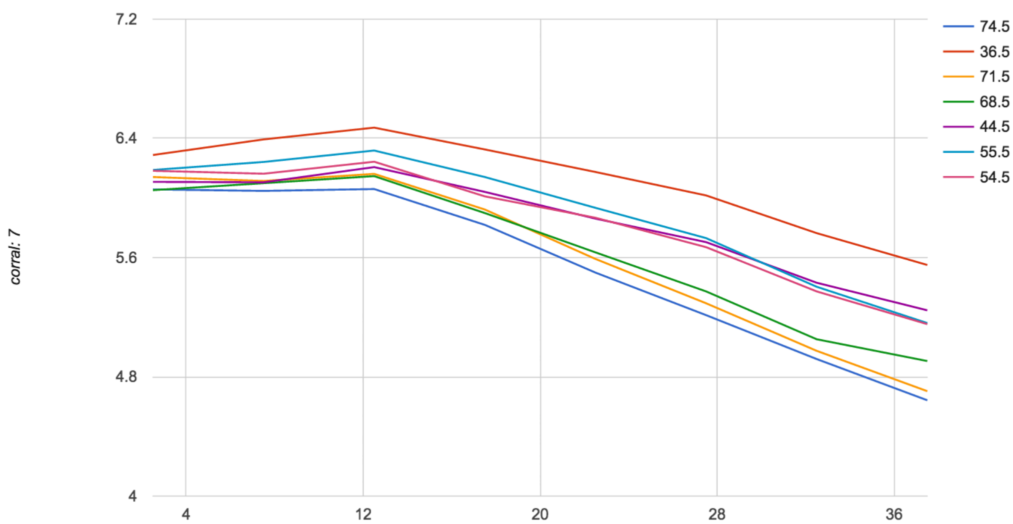
Additionally, we can have another graph displaying the error of our estimation, saying**:** <u>for a given 5k segment, our simulation suggests there are N number of people, and real-time data suggests M number of people, these people are most likely at mile X</u>**.**

To me, this (with some tweaks) would probably be the most reliable and trustworthy simulation we could use for calculating runner density. There would be room for improvement if we could somehow obtain access to the real-time tracking data, but even then we would only have 5k-segmented data and not mile-segmented information.

Corral C speed given position and temperature (F)



Corral F speed given position and temperature (F)



Runners per mile given minutes elapsed