Evil Genius x Genius League Assessment

2.A. Is entering via the light blue boundary a common strategy used by Team2 on T (terrorist) side?

 The data shows that Team2 entered through the light blue boundary a total of 5 rounds throughout the entire match. Knowing that Team2 only plays on T side for 15 rounds, it is pertinent to say that entering through the light blue region is a fairly common strategy as it happened in 33% of the total rounds.

2.B. What is the average timer that Team2 on T (terrorist) side enters "BombsiteB" with least 2 rifles or SMGs?

 The data shows that Team2 had at least 2 rifles/SMGs for 11 rounds. However, they only made it to the BombsiteB for 5 rounds out of the 11. For those 5 rounds, Team2's average time taken to get to the BombsiteB is 451470.4(ticks).

2.C. Now that we've gathered data on Team2 T side, let's examine their CT (counter-terrorist) Side. Using the same data set, tell our coaching staff where you suspect them to be waiting inside "BombsiteB"

 The heatmap displays some activities around the center and the top right area of the BombsiteB. There are some activities detected in the bottom right area, but it's not as active as the top right area.
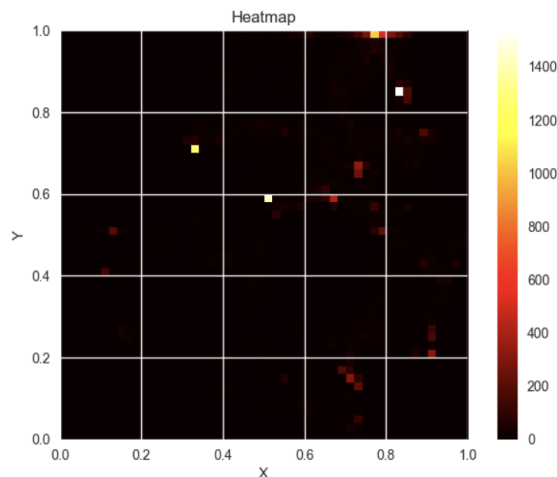
3. (No Coding) Most of the time, our stakeholders (in this case, the CS:GO coaching staff) aren't tech-savvy enough to run code themselves. Propose a solution to your product manager that:

a. could allow our coaching staff to request or acquire the output themselves

b. takes less than 1 weeks worth of work to implement

My first solution is a simple web application with Flask as the backend and React as the frontend. As a lightweight framework, flask should be easy to implement in a short period of time along with react. As a minimalistic application that only takes user input and displays the output, the web application wouldn't need many security measures, which would reduce the time to implement.

However, in order to implement this application in a timely manner, requirements need to be defined and clarified. The team must identify what do the coaching staff try to accomplish through coding, and what input and output they expect from the application. Depending on the scope of the project, completing the above solution in a week may not be feasible. In that case, we could implement a simple python program that lists the number of actions that users can take and enable them to specify their requests through the command line interface. In this way we could save time by not implementing a fully structured front end and still have backend logic for the coaching staff to work with.