Introduction/business problem:

In this project we worked towards finding an optimal location for a breakfast point. Specifically, this report will be targeted to stakeholders interested in opening a breakfast spot near richmond circle in Bangalore. Since there are lots of restaurants near richmond circle, we detected locations that are not already crowded with restaurants and eating joints. We were also particularly interested in areas with no breakfast spots in the vicinity. We preferred locations as close to richmond circle as possible, assuming that first two conditions are met. We used our data science powers to generate a few most promising neighborhoods based on this criteria. Advantages of each area will then be clearly expressed so that best possible final location can be chosen by stakeholders.

Data:

Factors that influenced our decisions are:

● Number of existing breakfast spots in the neighborhood

● Number of and distance to breakfast spots in the neighborhood, if any

● Distance of neighborhood from richmond circle We decided to use regularly spaced circular grids of locations, centered around Richmond Circle center, to define our neighborhoods. Following data sources were needed to extract/generate the required information:

● centers of candidate areas will be generated algorithmically and approximate addresses of centers of those areas will be obtained using Google Maps API reverse geocoding

● number of breakfast spots and their type and location in every neighborhood will be obtained using Foursquare API

● coordinate of center will be obtained using Google Maps API geocoding of well known Richmond Town in Bangalore.

Neighborhood Candidates:

We created latitude & longitude coordinates for centroids of our candidate neighborhoods. We will create a grid of cells covering our area of interest which is approx. 12x12 kilometers centered around Richmond Circle, Bangalore. We found the latitude & longitude of Richmond Circle, using specific, well known addresses and Google Maps geocoding API. We created a grid of area candidates, equally spaced, centered around city center and within ~3km from Richmond Circle. Our neighborhoods were defined as circular areas with a radius of 300 meters, so our neighborhood centers were 600 meters apart. Reason of choosing 3 km vicinity was because of the limit on calls that can be made to FourSquare API for number of generated neighborhoods. Only 950 calls / day can be done to Foursquare Api using a free developers account. To accurately calculate distances we needed to create our grid of locations in Cartesian 2D coordinate system which allows us to calculate distances in meters (not in latitude/longitude degrees). Then we projected those coordinates back to latitude/longitude degrees to be shown on Folium map. We created functions to convert between WGS84 spherical coordinate system (latitude/longitude degrees) and UTM Cartesian coordinate system (X/Y coordinates in meters). We created a hexagonal grid of cells: we offset every other row, and adjust vertical row spacing so that every cell center is equally distant from all it's neighbors. we now had the coordinates of centers of neighborhoods/areas to be evaluated, equally spaced (distance from every point to it's neighbors is exactly the same) and within ~3km from Richmond Circle.