Where to Eat in San Francisco?

1.Introduction

Restaurant recommendation apps such as Yelp or Foursquare provides plenty of information on restaurant food taste. However, for tourists unfamiliar with the city they are visiting, other than whether the food taste good or not, they might also want to get information on whether the restaurant may have potential health risks, whether the restaurant locates in a safe neighborhood and how difficult it is to find parking nearby, before they decide where to eat.

Here I'm using San Francisco as an example. Let's imaging a scenario that someone is visiting San Francisco and looking for a place for dinner. He has his family with little kids who are very sensitive to the food quality. Before he makes a decision where to eat, he wants to consider the food rating, health score, parking and crime rate.

2.Data:

I will explore the San Francisco restaurant data from Foursquare. They provide information such as name, address, location, latitude, longitude, ratings, tips of each restaurant. Which this information, I will be able to localize a list of restaurants surrounding the input location. Then I will explore the SF restaurant health score data sheet and crime data (downloaded from Kaggle) and match the restaurants with their average score. Both ratings and health score could be an indicator for restaurant selection. Finally, I will use the Foursquare location information again to explore the parking lots near the restaurant of choice.

3. Methodology

First step, address information is collected by input function, then converted into latitude and longitude information. With Foursquare API, I can get the search url for the type of restaurant I'm interested in. The output will be the search url.

```
please type in your address: 1200 Mission St, San Francisco
37.7724921 -122.4188648
What type of restaurant are you intereted in? : Japanese
Japanese .... OK!
```

Next, localize all the restaurant near the input location and create a data frame from the search result. Clean the column names and filter the category for each row. Only keep the most relevant information, such as name, categories, distance, etc.

	name	categories	address	distance	lat	lng	id
0	Ryoko's Japanese Restaurant & Bar	Sushi Restaurant	619 Taylor St	1851	37.788183	-122.411882	433c8000f964a52043281fe3
1	Cha-Ya Vegetarian Japanese Restaurant	Vegetarian / Vegan Restaurant	762 Valencia St	1324	37.760786	-122.421557	44ccee85f964a52016361fe3
2	Muracci's Japanese Curry & Grill	Japanese Curry Restaurant	307 Kearny St	2427	37.791000	-122.404282	49ba1813f964a52059531fe3
3	Miyabi Japanese Restaurant	Sushi Restaurant	253 Church St	1085	37.766829	-122.428903	43a8315cf964a520742c1fe3
4	Japanese YWCA Building / Nihonmachi Little Fri...	Nursery School	1700 Sutter St	1918	37.787017	-122.430609	4d2ca3774ab78eec71103326

Then, import the csv with restaurant scores downloaded from Kaggle. Multiple inspections were done with the same restaurant, so I used a group by for name, and calculated the mean score for each restaurant. I did an inner join to create a form with restaurant score added. I also used for loop to search for restaurant rating for each specific restaurant in the joint form from Foursquare API. Here is the output of the form:

	name	categories	address	distance	lat	lng	id	inspection_score	rating
0	Japanese House	Japanese Restaurant	480 6th St	1457	37.776341	-122.403028	5a3adef8ca18ea1428d66942	82.0	5.5
1	An Japanese Restaurant	Sushi Restaurant	22 Peace Plz #510	1663	37.785029	-122.429155	563ba07ccd10e452f21ab439	98.0	7.9
2	Akira Japanese Restaurant	Restaurant	1634 Bush St	1813	37.788234	-122.424170	585cab80bbec66033bb4b92f	88.0	8.0
3	Rock Japanese Cuisine	Japanese Restaurant	Pine St	2415	37.791562	-122.405772	57a16052498ef2fd3cadd5a3	85.0	5.7
4	Niji Japanese Grille	Noodle House	50 Post St	2348	37.789393	-122.402882	4a0260bbf964a5204b711fe3	86.0	5.1

Next, I imported the crime rate data downloaded from Kaggle. I used KDTree method to count the number of crimes within 300 meters of each restaurant. Then appended the result to the data frame.

	name	categories	address	distance	lat	lng	id	inspection_score	rating	Crime_Incidents
0	Japanese House	Japanese Restaurant	480 6th St	1457	37.776341	-122.403028	5a3adef8ca18ea1428d66942	82.0	5.5	1634
1	An Japanese Restaurant	Sushi Restaurant	22 Peace Plz #510	1663	37.785029	-122.429155	563ba07ccd10e452f21ab439	98.0	7.9	1706
2	Akira Japanese Restaurant	Restaurant	1634 Bush St	1813	37.788234	-122.424170	585cab80bbec66033bb4b92f	88.0	8.0	1200
3	Rock Japanese Cuisine	Japanese Restaurant	Pine St	2415	37.791562	-122.405772	57a16052498ef2fd3cadd5a3	85.0	5.7	2388
4	Niji Japanese Grille	Noodle House	50 Post St	2348	37.789393	-122.402882	4a0260bbf964a5204b711fe3	86.0	5.1	2673

Then, I explored the parking options within 300 meters of the restaurant and counted the total number of parking. Then I appended the column: Parking_Count.

	name	categories	address	distance	lat	lng	id	inspection_score	rating	Crime_Incidents	Parking_Count
0	Japanese House	Japanese Restaurant	480 6th St	1457	37.776341	-122.403028	5a3adef8ca18ea1428d66942	82.0	5.5	1634	9.0
1	An Japanese Restaurant	Sushi Restaurant	22 Peace Plz #510	1663	37.785029	-122.429155	563ba07ccd10e452f21ab439	98.0	7.9	1706	3.0
2	Akira Japanese Restaurant	Restaurant	1634 Bush St	1813	37.788234	-122.424170	585cab80bbec66033bb4b92f	88.0	8.0	1200	5.0
3	Rock Japanese Cuisine	Japanese Restaurant	Pine St	2415	37.791562	-122.405772	57a16052498ef2fd3cadd5a3	85.0	5.7	2388	21.0
4	Niji Japanese Grille	Noodle House	50 Post St	2348	37.789393	-122.402882	4a0260bbf964a5204b711fe3	86.0	5.1	2673	23.0

Before I create a demo algorithm, I need to convert all data into the same scale. So I used the ratio to max method and get the ratio of each column to the max value of the specific column. Here is the converted dataframe.

	name	categories	address	distance	lat	lng	id	inspection_score	rating	Crime_Incidents	Parking_Count
0	Japanese House	Japanese Restaurant	480 6th St	1457	37.776341	-122.403028	5a3adef8ca18ea1428d66942	0.836735	0.6875	0.611298	0.391304
1	An Japanese Restaurant	Sushi Restaurant	22 Peace Plz #510	1663	37.785029	-122.429155	563ba07ccd10e452f21ab439	1.000000	0.9875	0.638234	0.130435
2	Akira Japanese Restaurant	Restaurant	1634 Bush St	1813	37.788234	-122.424170	585cab80bbec66033bb4b92f	0.897959	1.0000	0.448934	0.217391
3	Rock Japanese Cuisine	Japanese Restaurant	Pine St	2415	37.791562	-122.405772	57a16052498ef2fd3cadd5a3	0.867347	0.7125	0.893378	0.913043
4	Niji Japanese Grille	Noodle House	50 Post St	2348	37.789393	-122.402882	4a0260bbf964a5204b711fe3	0.877551	0.6375	1.000000	1.000000

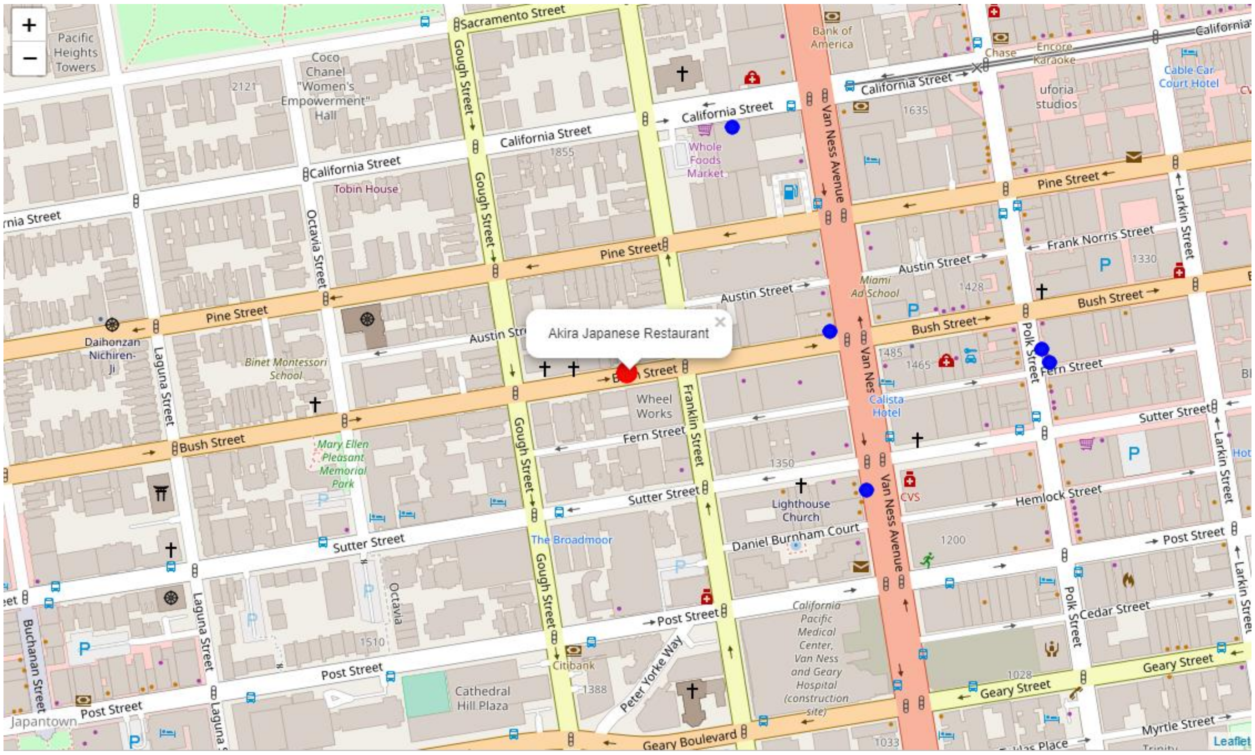
The demo algorithm I'm using here is:
rs=(0.2*dfj2.inspection_score+0.55*dfj2.rating+0.15*dfj2.Parking_Count+0.1*1/dfj2.Crime_Incidents)*100

4.Result

Below is the output form, bar graph based on the demo algorithm.

name	lat	lng	Recommendation_Score
Japanese House	37.776341	-122.403028	77.0
An Japanese Restaurant	37.785029	-122.429155	92.0
Akira Japanese Restaurant	37.788234	-122.424170	98.0
Rock Japanese Cuisine	37.791562	-122.405772	81.0
Niji Japanese Grille	37.789393	-122.402882	78.0

This is a map showing the selected restaurant and parking lots nearby.



5. Discussion

Here I'm using a demo algorithm to calculate the final output score. In the real-world scenario, how each component is weighed should be based on individual preferences. Questionnaires should be given to collect the preferences and serve as basis for the coefficients.

This is a very preliminary project. To dig deeper, I'd like to get information for actual parking availability, such as number of spaces for each parking lot and occupancy rate. This will added to further develop this project.