Opening a Pure Veg Restaurant Business near my locality

Data Science Project

Introduction:

Growing as a foodie, I wanted to always start a restaurant business near my locality (AMB Cinemas – Hyderabad, India). Restaurants have been growing exponentially over the last decade with variety of cuisines available near doorstep. Hyderabad, India has been the IT Hub for quite sometime resulting in influx of huge population of younger population into the city. I see this as an opportunity for me start my business and venture into the world of business.

Business Problem:

I want to open of a Pure Vegetarian Indian dining restaurant in Hyderabad, India near my locality AMB Cinemas Mall. My locality is densely populated with multiple restaurants with various cuisines. Due to cut-throat competition, there has been a decline in quality of food served and hence I want to use Data Science to understand whether opening a quality Pure-Veg Restaurant is a profitable venture or not. I will analyze few parameters available to arrive at my conclusion.

Communities/people that will benefit:

- Any Entrepreneur who wants to venture into restaurant business or who is planning to start any new business, or This provides a framework that can be generalized
- Businessmen who would want to expand their restaurant chains into newer locality

Data Section

Primarily, I want to collect list of all restaurants in and around my location - AMB Cinemas (~5KM Radius).

Secondarily, I will be using Four Square API to collect the list, cleanse and filter the data using data frame to build the necessary data set.

Finally, I will use Folium Map Model to analyze the density of vegetarian restaurants of collect few more parameters to arrive at the conclusion

Gather the information from the Four Square:

```
'https://api.foursquare.com/v2/venues/search?client_id=LAAFIKWSIOAZLUJXOTYGS3T4WQJCCFSX3FO3NMIL00BX5BNB&cl
ient_secret=ZDSPHLMEEMDA5XKGGWTYXUT35KYLGR4RPLBXNLHYGQ1TUHNT&ll=17.457321,78.3634764&v=20180604&query=Rest
aurant&radius=4500&limit=50'
```

```
results = requests.get(url).json()
results
```

Transformation of Information:

```
# assign relevant part of JSON to venues
venues = results['response']['venues']

# tranform venues into a dataframe
dataframe = json_normalize(venues)

dataframe.head(100)
```

```
/home/jupyterlab/conda/envs/python/lib/python3.6/site-packages/ipykernel_launcher.py:5: FutureWarning: pan
das.io.json.json_normalize is deprecated, use pandas.json_normalize instead
  """
```

	id	name	categories	referralId	hasPerk	location.lat	location.lr
0	4dc404efd164eb9c9fca0e8c	Gardenia Bar & Restaurant	[{'id': '4bf58dd8d48988d117941735', 'name': 'B...	v-1586430804	False	17.459301	78.366470
1	5bdbfdfd1a2925003247dd49	Swadesh Multi-cuisine Restaurant	[{'id': '54135bf5e4b08f3d2429dfe0', 'name': 'M...	v-1586430804	False	17.457851	78.365844
2	515d384ce4b0d7a8c1412e28	Mirch Masala Restaurant	[{'id': '4bf58dd8d48988d10f941735', 'name': 'I...	v-1586430804	False	17.455738	78.366293

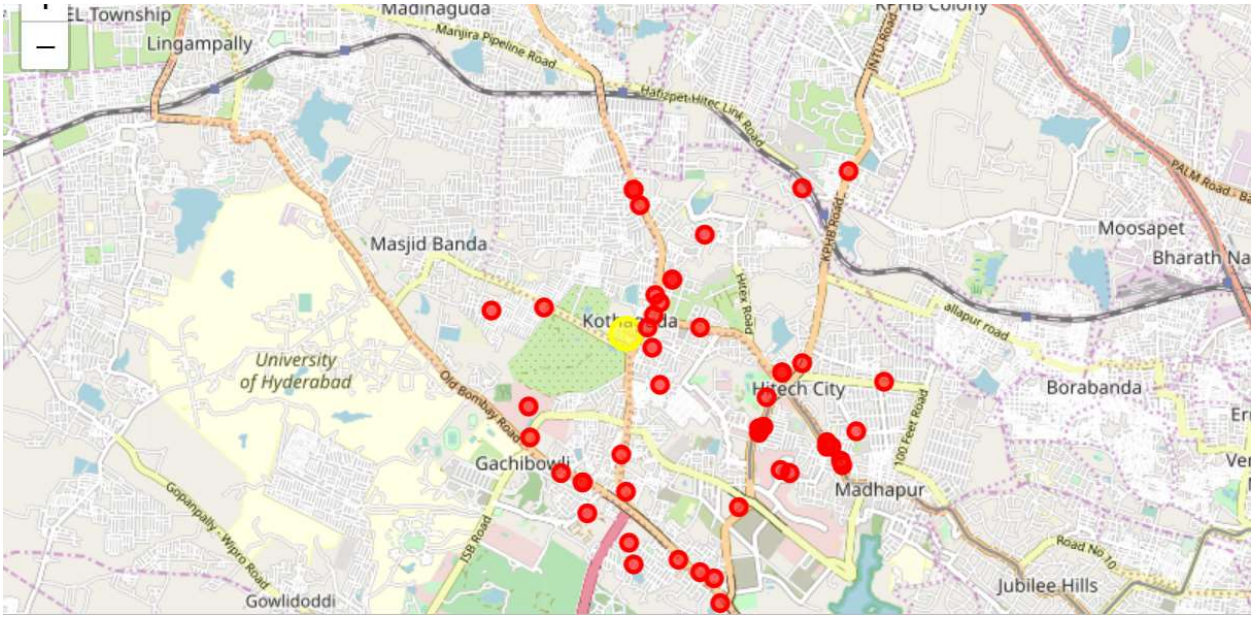
Filtering of the information:

Define information of interest and filter dataframe

```
# keep only columns that include venue name, and anything that is associated with location
filtered_columns = ['name', 'categories'] + [col for col in dataframe.columns if col.startswith('locatio
n.')] + ['id']
dataframe_filtered = dataframe.loc[:, filtered_columns]

# function that extracts the category of the venue
def get_category_type(row):
    try:
        categories_list = row['categories']
    except:
        categories_list = row['venue.categories']

    if len(categories_list) == 0:
        return None
    else:
        return categories_list[0]['name']

# filter the category for each row
dataframe_filtered['categories'] = dataframe_filtered.apply(get_category_type, axis=1)

# clean column names by keeping only last term
dataframe_filtered.columns = [column.split('.')[-1] for column in dataframe_filtered.columns]

#Build a new data frame with filtered Columns
df_new = dataframe_filtered[['name','categories','lat','lng','distance','postalCode','id']]
df_new['likes']          = 0
df_new['rating']         = 0
df_new['type']           = ''

df_new["categories"]     = df_new["categories"].str.upper()
df_new
```

Density Display:

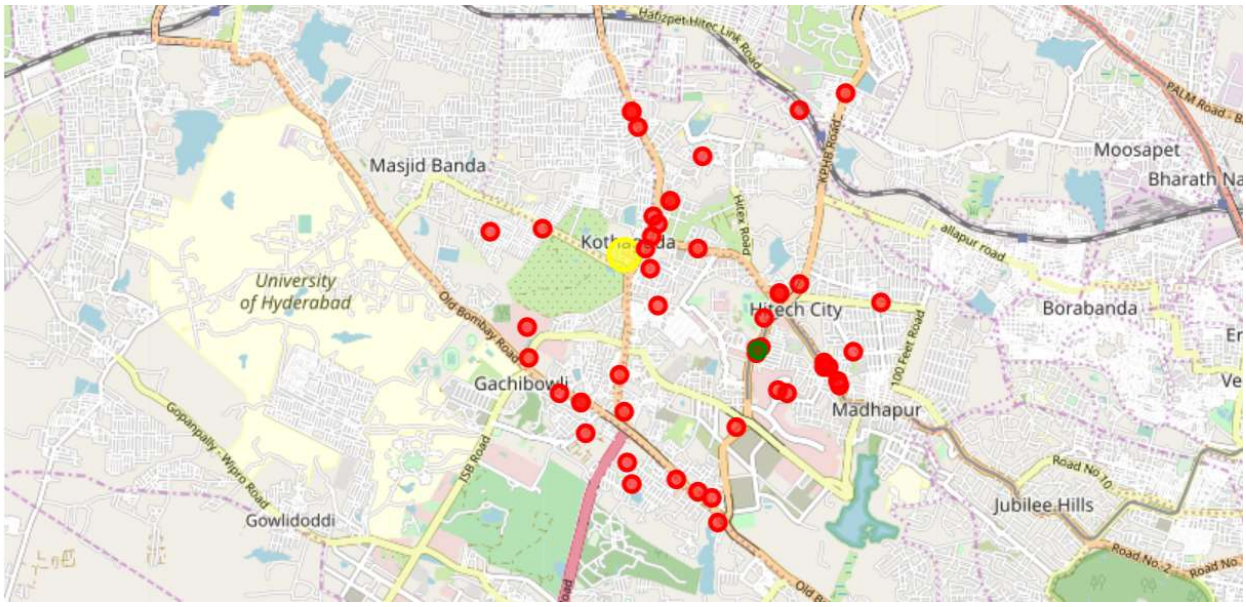
Density Map of all the restaurants around 5 KM Radius – Limit 50



Get the list of Pure Veg list and mark them with Green Color:

```
#Filter Based on Vegan or Pure Veg Resturants
df_fil_veg             = df_new
df_fil_veg['Pure Veg'] = df_new['categories'].str.contains('VEGAN')
#Build the Index
df_pure_veg = df_fil_veg[(df_fil_veg['Pure Veg'] == True)]

df_pure_veg

## Add this to the existing Map with a different colour

# add the Italian restaurants as blue circle markers
for lat, lng, label in zip(df_pure_veg.lat, df_pure_veg.lng, df_pure_veg.categories):
    folium.features.CircleMarker(
        [lat, lng],
        radius=5,
        color='Green',
        popup=label,
        fill = True,
        fill_color='Green',
        fill_opacity=0.6
    ).add_to(venues_map)
```

Density Display:



Additional Parameters of the restaurants (Likes Count and Rating):

```
#Loop Through Each Resturant and find the no of likes for that restuarant
for i in range(len(df_new)) :
    venue_id = df_new.loc[i,"id"]
    url = 'https://api.foursquare.com/v2/venues/{}?client_id={}&client_secret={}&v={}'.format(venue_id, CLIENT_ID,
    result = requests.get(url).json()
    try:
        df_new.loc[ i, "likes"]  = result['response']['venue']['likes']['count']
    except:
        df_new.loc[ i, "likes"]  = 0
    try:
        df_new.loc[ i, "rating"] = result['response']['venue']['rating']
    except:
        df_new.loc[ i, "rating"]  = 0
    try:
        df_new.loc[ i, "type"] = result['response']['venue']['categories'][0]['pluralName']
    except:
        df_new.loc[ i, "type"] = 'Multi-Cuisine'
df_new
```

name	categories	lat	lng	distance	postalCode	id	likes	rating	type	Veg
Gardenia Bar & Restaurant	BEER GARDEN	17.459301	78.366470	386	500084	4dc404efd164eb9c9fca0e8c	1	0.0	Beer Gardens	False
Swadesh Multi-cuisine Restaurant	MULTICUISINE INDIAN RESTAURANT	17.457851	78.365844	258	500084	5bdbfdfd1a2925003247dd49	0	0.0	Multicuisine Indian Restaurants	False
Mirch Masala Restaurant	INDIAN RESTAURANT	17.455738	78.366293	347	NaN	515d384ce4b0d7a8c1412e28	1	0.0	Indian Restaurants	False
Teja Multicuisine Restaurant	None	17.460486	78.367143	525	500084	518e6f1d498e1abaef4e8642	0	0.0	Multi-Cuisine	None
Athidhyam Restaurant	DINER	17.461315	78.366654	558	NaN	539b1989498ef1b1c9bd60ca	0	0.0	Diners	False
Hyderabad Restaurant	INDIAN RESTAURANT	17.443802	78.387292	2942	500081	502d1288e4b0f9a2dc332f18	1	0.0	Indian Restaurants	False
Teja Multicuisine Restaurant	None	17.451923	78.367143	716	500084	518e6fb5498ebdadd17f71d2	0	0.0	Multi-Cuisine	None

Analysis & Conclusion:

From the above, we can understand that there is a huge density of restaurants around the AMB Cinema locality in Hyderabad, India. But as we can observe that there are very few Pure-Veg Restaurants around our locality. We can also observe that the restaurants do not have a good rating and like counts have been considerably low.

This emphasizes that the quality of the existing restaurants is considerably low and with very few Pure-Veg restaurants available, I would like to conclude that opening a **Pure – Veg restaurant at my locality (AMB Cinemas, Hyderabad, India) would be a profitable venture.**