A Project/Dissertation ETE REVIEW

on

Geolocation Location Analysis
Submitted in partial

fulfillment of the requirement

for the award of the degree of

Bachler of Technology, School of Computer Science and Engineering

Under The Supervision of

Mr. Surendra Singh Chauhan

Submitted By

Rohit Sahu 19021011417

Anubhav Soni 19SCSE1010693

SCHOOL OF COMPUTING SCIENDE AND ENGINEERING DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING GALGOTIAS UNIVERSITY, GREATER NOIDA

INDIA

MONTH, YEAR

SCHOOL OF COMPUTING SCIENCE AND ENGINEERING

GALGOTIAS UNIVERSITY, GREATER NOIDA

CANDIDATE'S DECLARATION

I/We hereby certify that the work which is being presented in the thesis/project/dissertation, entitled **"Geolocation analysis for suitable "** in partial fulfillment of the requirements for the award of the_____submitted in the School of Computing Science and Engineering of Galgotias University, Greater Noida, is an original work carried out during the period of month, Year to Month and Year, under the supervision of Mr. Surrendra Singh Chauhan, Assistant Professor, Department of Computer Science and Engineering, Galgotias University, Greater Noida

The matter presented in the thesis/project/dissertation has not been submitted by me/us for the award of any other degree of this or any other places.

Rohit Sahu, 19SCSE1010228

Anubhav Soni, 19SCSE1010693

This is to certify that the above statement made by the candidates is correct to the best of my knowledge.

Mr. Surendra Singh Chauhan

Assistant Professor

CERTIFICATE

The Final Thesis/Project/ Dissertation Viva-Voce examination of Name: Admission No has been held on _____an

dhis/her work is recommended for the award of

Signature of Examiner(s)

Signature of Supervisor(s)

Signature of Project Coordinator

Date:

Place: Greater Noida

student Information

Name	Admission Number	Enrolment Number	Program/Branch	Section/Batch
Rohit Sahu	19SCSE1010228	19021011417	B.Tech/SCSE	03/p2
Anubhav Soni	19SCSE1010693	19021011841	B.Tech/SCSE	03/p1 ‘

Faculty Information

Name	Designation	Role
Mr. Surendra Singh Chauhan	Assistant Professor	Guide
Mr. Dhruv Kumar	Assistant Professor	Ete Reviewer

Table of figures

Acronyms

PG	Paying Guest
B.Tech	Bachler of Technology
API	Application Programming Interface
SCSE	School of Computer Science and Engineering
REST	Representational State Transfer

Abstract

It happens too often that one has to travel to another unknown city, for work, education, or tourism, so it becomes that one findsplace to accommodate, now if you are going on a vacation then it is short term stay, you can just book a hotel and do the side seeing. For student, immigrant or worker. They want affordable place to stay.

This project tries to find the best possible location according to some predetermined parameters which are set and trained according to the data collected from such type of people, and Geolocation of a particular location can be used, then the data willbe fed to the K-Means Clustering algorithm, whose result will then be represented on a Map, using REST API.

For this purpose the Machine Learning algorithm, K-Means Clustering, will be used to find the locations or cluster of locations. Initially data will be read, using Python's Pandas library, and this data will be cleaned using Pandas as well, then the data will bevisualized using matplotlib library or pandas, to gain insights from the cleaned data. For presenting the Geolocation data REST API will be used.

The final result will be presented on a Map using REST API, which will make it much easier to locate the desired location or getthe approximate idea to the final location. This analysis will greatly narrow down the search for the place to stay

Lastly it is to be expected that the data will be needed to do the such analysis, and the result are profitable for the businesses, LikeApartments, Hostels, or Paying Guest Houses. Small business owner can use this data to find the ideal location for their new venture.

KEY WORD: Machine Learning, suitable accommodation.

Chapter-1: Introduction

1.1. This project tries to look into the problem and hassle of finding a suitable place to stay for immigrant, Job seekers orstudent. And it does so using collected data of preferences and taste of individual and Geolocation data to show where is that ideal place.

1.2. It is not a uncommon that a certain individual need to travel from his/her home town to a town, where he has to be there in order to do some kind of work, they are migrating for better quality of life or their job takes them there. Sometimes on short notice you have to travel, so there is very little time to plan, the traveller doesn't have someoneknown in the said city then it becomes new experience to find a place on your own when you have to manage your job, or immigration process, any other thing that you have arrived to do.
Also, when business owner want to venture into some idea that involves choosing a location, then it is imperative tohave appropriate information about the customer needs and location details.

1.2.1
K-means Clustering
K-means clustering (MacQueen, 1967) is a method commonly used to automatically

partition a data set into kgroups. It proceeds by selecting k initial cluster centers and then iteratively refining them as follows:
The algorithm converges when there is no further change in assignment of instances to clusters. In this work, we initialize the clusters using instances chosen at random from the data set. The data sets we used are composed solelyof either numeric features or symbolic features. For numeric features, we use a Euclidean distance metric; for symbolic features, we compute the Hamming distance. The final issue is how to choose k.[1]

Python
Python is PROGRAMMING LANGUAGE that lets you work quickly and integrate systems more effectively.[3]

Pandas

pandas is a fast, powerful, flexible and easy to use open source data analysis and manipulation tool, built on top of the Python programming language. [4]

Matplotlib

Matplotlib is a comprehensive library for creating static, animated, and interactive visualizations in Python.Matplotlib makes easy things easy and hard things possible. [5]

Seaborn

Seaborn is a Python data visualization library based on matplotlib. It provides a high-level interface for drawingattractive and informative statistical graphics. [6]

Folium

Folium is a python Library that that builds on the data wrangling strengths of the Python ecosystem and the mappingstrengths of the leaflet.js library. Manipulation of data is done in python, then visualize it in on a leaflet map via folium. [7]

3. The Whole Process at a glance. The steps that involved are:

1. Data collection, [8]
2. Data cleaning,
3. Visualizing the data,
4. Applying the algorithm,
5. Get geolocation data from foursquare API,
6. Presenting the results on the folium library.

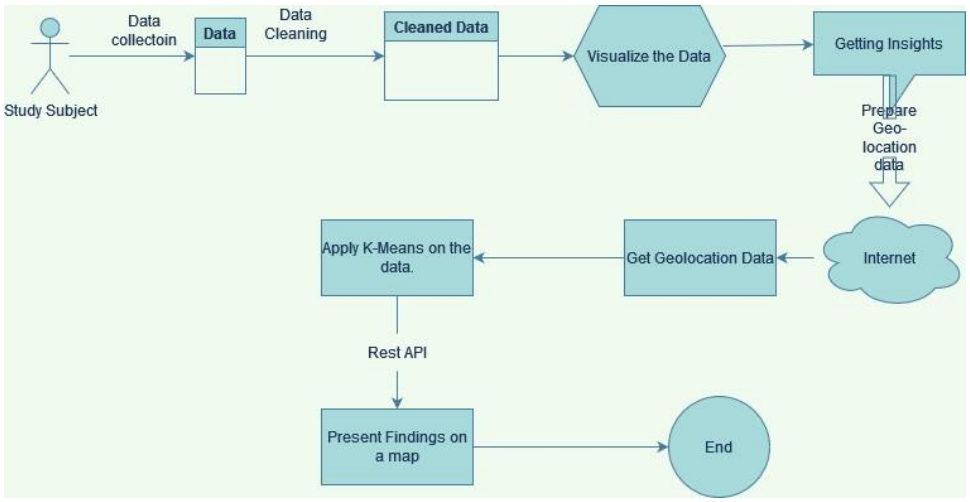


FIG: Steps

Chapter-2: Literature Survey

When one has to find a place to live in (not your hometown), the easiest way is to ask a group of people and they maybe will tellyou the place you desire or somewhat match your description of your ideal place, but people usually tell from their experience which is based on their likes and dislikes.

This process is often time consuming might not meet your requirements. We also want to look at this problem at the business point of view, the revenue of a business depends on how many people are using their product or their service, and if they want tomaximize their profit then they want as many people buy their product as much possible.

So, it would make sense that would want in on the information that t

This research is useful to businesses and working individuals alike. As this analysis give knowledge about where is the density ofworking individual, the business owner can decide where to move or open their business, this could lead to hike in profit margin,and overall positive growth.

Individuals can find a place to stay in unknown city or in their own city if they want to move somewhere within the city. Thisproject can be scaled from localities, to cities to country.

Pandas

Chapter 3: Explaining the process

pandas is a fast, powerful, flexible and easy to use open source data analysis and manipulation tool, built on top of the Python programming language. [4]

1. **Data collection**

The data is obtained from Kaggle [8]
The geolocation data is obtained from the Foursquare API, the location data is obtained around Galgotias University, within a radius of 30 km, as the University is remote and there are very few places to stay, within 10 km, that suit the budget of your normal student.
Then data is loaded using the panda's library.

2. **Data cleaning**

Initially the data contains about 60 features, most of them are irrelevant and some are not numerical so they need to be removed.

The features that stay are, ['cook', 'eating_out', 'employment', 'ethnic_foo'
, 'exercise', 'fruit_day', 'income',
'on_off_campus', 'parents_cook',
'paymeal_out', 'sports', 'veggies_day']
Total features = 12
These features have multiple or two options that are number from 1(least likely) to 6(most likely) According to the question answer change.

3. Visualize the data

The cleansed data is then plotted, using box plot This how the data looks

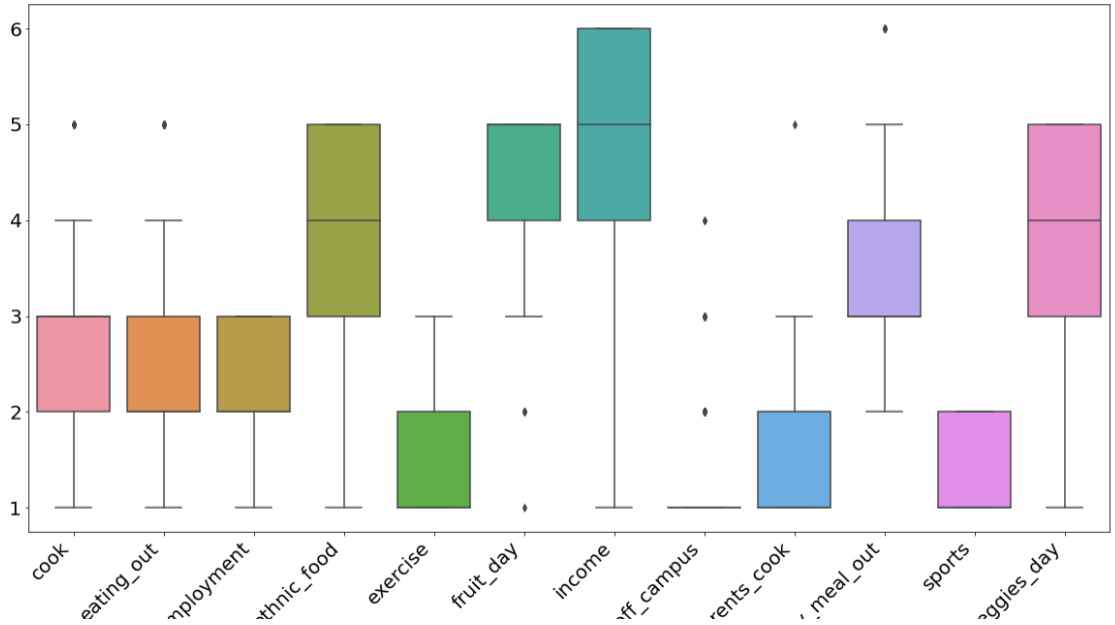


Fig: Data Visualization

4. Apply k-means clustering on the user data and get insights

A intrinsic step for any unsupervised algorithm is to determine the optimal number of clusters into which the given data is or may be clustered.[12]

To the do exactly this there is a method called 'The Elbow Method'[11]. This method is very popular to determine the optimal number of clusters, that the data may be clustered in .

This method uses the SSE or Sum of Squared Distance between the data points and their respective assigned clusters centroid or says mean value. And we pick k value at where the point SSE starts to flatten out and forming an elbow.[10]

This is how the method helps to find the good value of k (number of clusters for the dataset) and help in making the good clusters for the given dataset.

The data set for run on 1 to 20 clusters and using the Kmeans algorithm and then the result of those cluster was plotted on a graph. This result look like this:

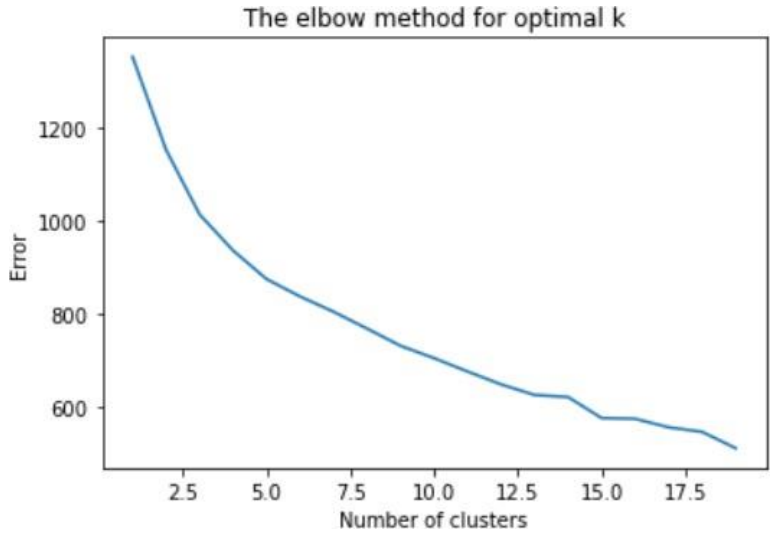


Fig: Elbow Method, for optimal k value

The optimal value chosen for the algorithm to apply to is 3. We noticed that when the clusters were increased or decreased then the result were not very useful so we had to choose 3, as the number of clusters.

Then we will apply the actual algorithm to the data and plot the clusters using matplotlib and find some trend.

K-Means clustering is applied with three clusters, this is how they look.

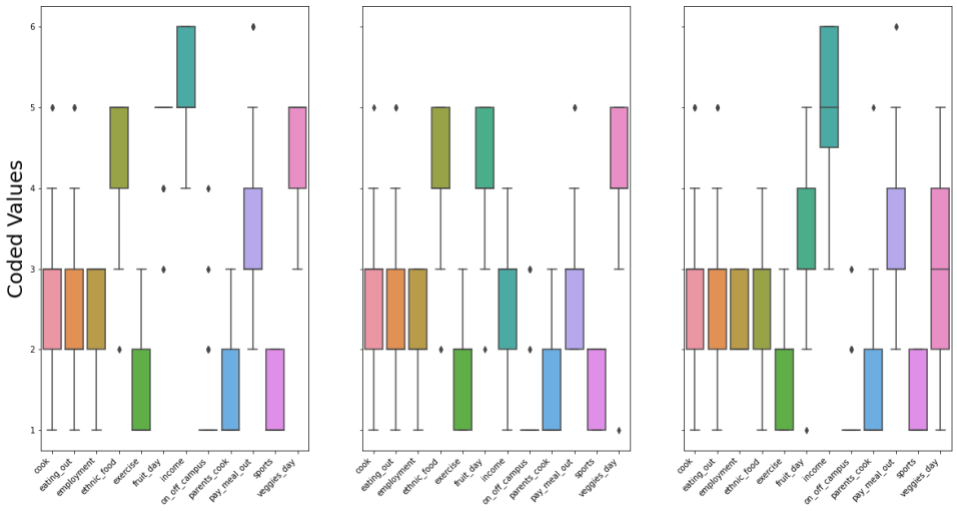


Fig: The clusters in the data

There are some points that can be noticed,

As the income increases

People with high income:
- Prefer fruit day
- Prefer to go out and eat at a restaurant sometimes.

People with low income:

- Prefer to stay home and cook
- Rarely want to go out and eat.
- Want to have vegetable shops around them.

5. Get geolocation data from Foursquare API [5]
   To get the venues data from foursquare API visit their website.
   Register on the website and login.[5]
   Find the Places and Pilgrim SDK and log in there.[5]
   You will be taken to your projects page.
   Make a new project there, and you will be given “Client id” and “Client Secret”, what will be used to, access the access the data.

6. According to the insights present location data on folium

The result is divided according to the distance,

1. blue marker is my college
2. green are the location between 10 and 20 kilomter
3. orange are the distance between 20 and 30 kilometer
4. red are the distance between 30 to 35 km
5. darkred are the distance greater than 35(not at all feasable and no location given)

6. pink could have the distance below 10 kilometer but data could not show that, but there are two location slightly greater than 10(green).

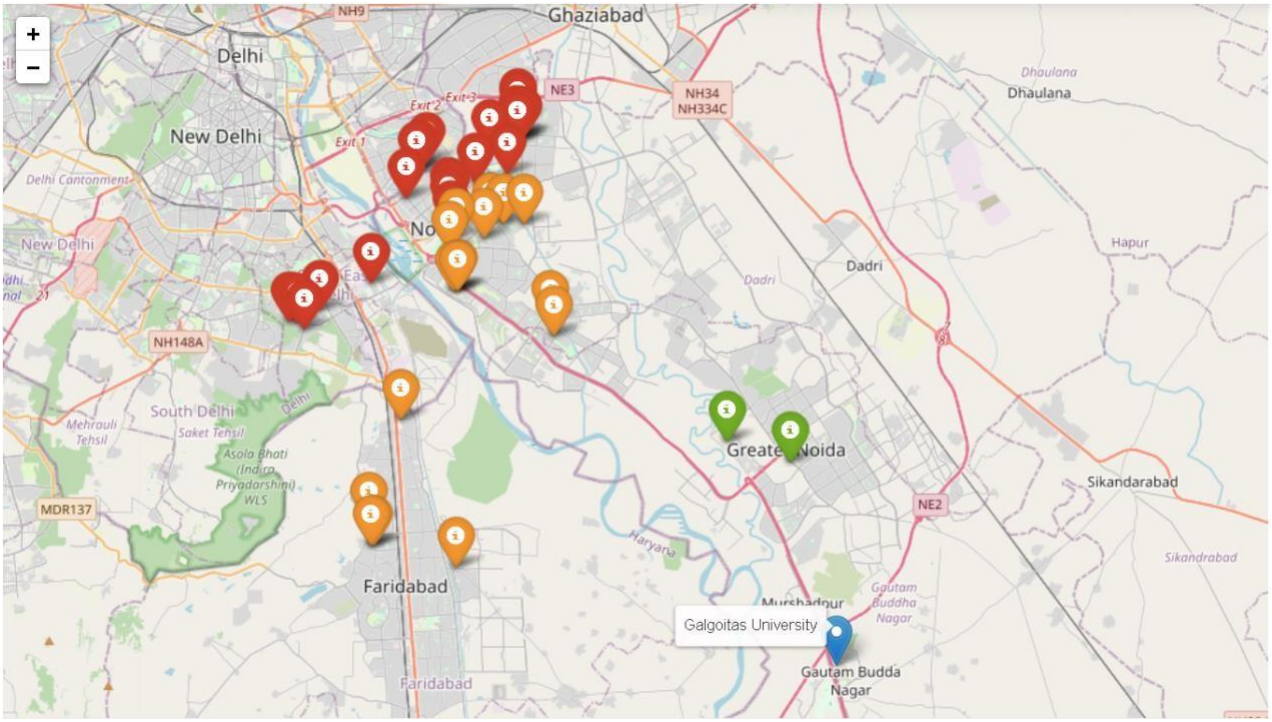


Fig: Result

Chapter 4: Result and Discussion

The data that we got around the GU, is not very good, as the location are spread far apart, and the nearest location found was roughly 10 kilo meters away.

There are location that are between 20 to 30 kilometer but there are not that many and as the distance increase the time that is taken to get to the university is also increased.

When the places such as restaurant and grocery market, were found out using the foursquare API around that location then the number was above 5 or 10 that is more than enough for a student or a Service worker, as there are more residential areas, then the services available also increase, the problem comes when there are less people, that usually means that the business owners would not want to move to that location and that are there then to hike the price of the commodities, in order to earn more thus make more profit.

Chapter 5

Project Execution

12

```
import requests # library to handle requests
import pandas as pd # library for data analsysis
import numpy as np # library to handle data in a vectorized manner
import random # library for random number generation
import matplotlib.cm as cm
import folium
from sklearn.cluster import KMeans
import matplotlib.pyplot as plt
import seaborn as sns

# tranforming json file into a pandas dataframe library
from pandas.io.json import json_normalize

%matplotlib inline
```

Import data set

```
df = pd.read_csv('food_coded.csv')
```

Removing all the features that are not needed.

all the non numerical, and features that have little effect on,

deciding a place to live, are removed.

```
droping = ['GPA',
 'Gender',
 'breakfast',
 'calories_chicken',
 'calories_day',
 'calories_scone',
 'coffee',
 'comfort_food',
 'comfort_food_reasons',
 'comfort_food_reasons_coded',
 'comfort_food_reasons_coded.1',
 'cuisine',
 'diet_current',
 'diet_current_coded',
 'drink',
 'eating_changes',
 'eating_changes_coded',
 'eating_changes_coded1',
 'father_education',
 'father_profession',
 'fav_cuisine',
```

```
 'fav_cuisine_coded',
 'fav_food',
 'food_childhood',
 'fries',
 'grade_level',
 'greek_food',
 'healthy_feeling',
 'healthy_meal',
 'ideal_diet',
 'ideal_diet_coded',
 'indian_food',
 'italian_food',
 'life_rewarding',
 'marital_status',
 'meals_dinner_friend',
 'mother_education',
 'mother_profession',
 'nutritional_check',
 'persian_food',
 'self_perception_weight',
 'soup','thai_food',
 'tortilla_calories',
 'turkey_calories',
 'type_sports','vitamins',
 'waffle_calories',
 'weight']
df3 = df.drop(droping, axis=1)
df3.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 125 entries, 0 to 124
Data columns (total 12 columns):
 #   Column         Non-Null Count  Dtype  
---  ------         --------------  -----  
 0   cook           122 non-null    float64
 1   eating_out     125 non-null    int64  
 2   employment     116 non-null    float64
 3   ethnic_food    125 non-null    int64  
 4   exercise       112 non-null    float64
 5   fruit_day      125 non-null    int64  
 6   income         124 non-null    float64
 7   on_off_campus  124 non-null    float64
 8   parents_cook   125 non-null    int64  
 9   pay_meal_out   125 non-null    int64  
 10  sports         123 non-null    float64
 11  veggies_day    125 non-null    int64  
dtypes: float64(6), int64(6)
memory usage: 11.8 KB
```

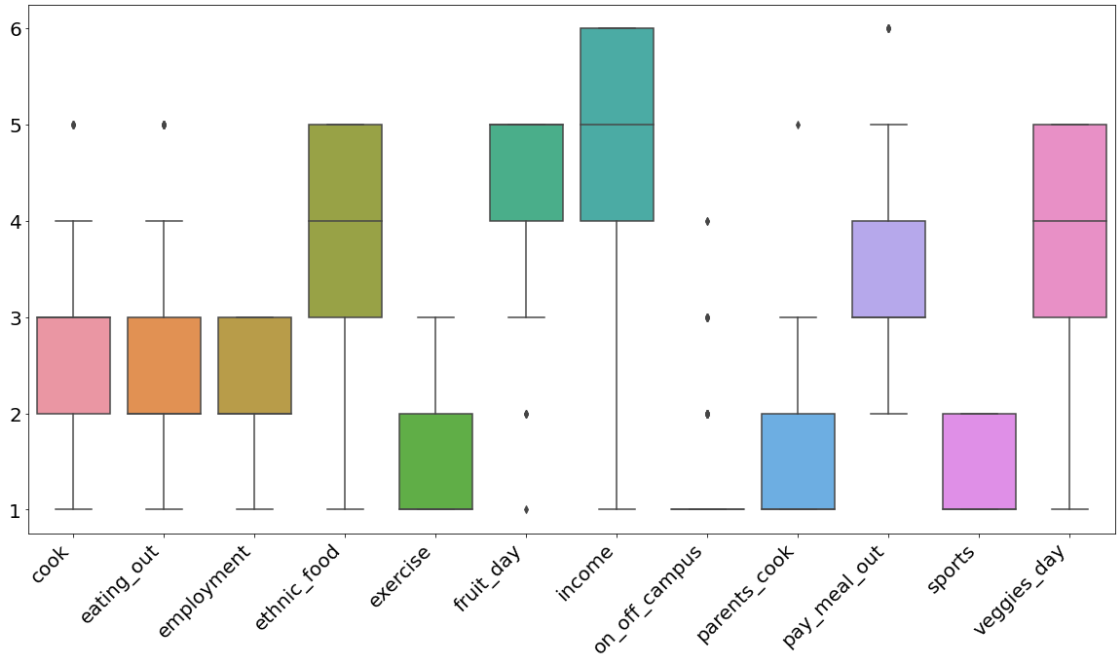
Replacing all the null values with the current mean of the features

```
a = (0, 0, 0, 0, 0, 0)
a = df3['employment'].mean(), df3['exercise'].mean(),df3['income'].mean(),
df3['on_off_campus'].mean(), df3['sports'].mean(), df3['cook'].mean()
a = pd.Series(a).apply(lambda x : int(x))
for i, col in enumerate(['employment', 'exercise', 'income', 'on_off_campus',
'sports', 'cook']):
    df3[col].fillna(a[i], inplace=True)

df3.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 125 entries, 0 to 124
Data columns (total 12 columns):
 #   Column         Non-Null Count  Dtype
---  ------         --------------  -----
 0   cook           125 non-null    float64
 1   eating_out     125 non-null    int64
 2   employment     125 non-null    float64
 3   ethnic_food    125 non-null    int64
 4   exercise       125 non-null    float64
 5   fruit_day      125 non-null    int64
 6   income         125 non-null    float64
 7   on_off_campus  125 non-null    float64
 8   parents_cook   125 non-null    int64
 9   pay_meal_out   125 non-null    int64
 10  sports         125 non-null    float64
 11  veggies_day    125 non-null    int64
dtypes: float64(6), int64(6)
memory usage: 11.8 KB
```

Visualizing using the 'boxplot' of the seaborn library.

```
plt.figure(figsize=(20, 10))
sns.boxplot

ax = sns.boxplot(data = df3)
ax.tick_params(labelsize=20)
plt.xticks(rotation=45, ha='right')
plt.savefig('cleaned.png')
plt.show()
```



it seems that people prefer to eat out less for low income, and high income eat sometimes

preference of ethinc food seems to increased with the income

people eat more fruits with more income(according to data at least)

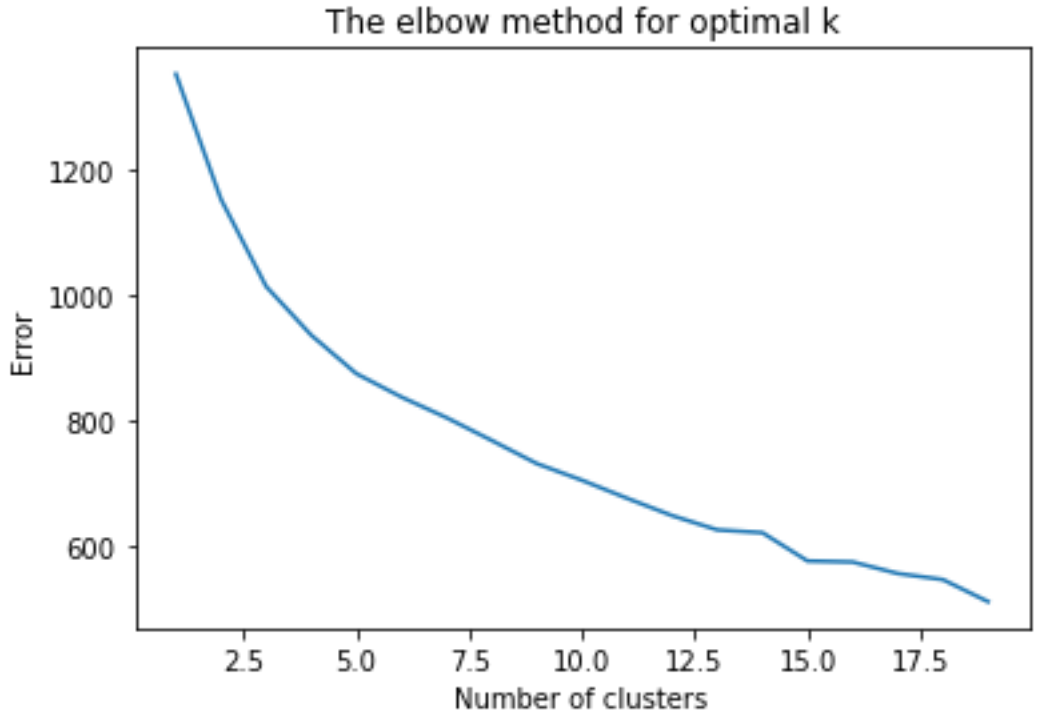
Apply the K-Means algorithm

Using Elbow method to decide the number of clusters.

```
wcss = []

for i in range(1, 20):
    kmeans = KMeans(n_clusters=i, init="k-means++", max_iter=300, n_init=10,
random_state=0)
    kmeans.fit(df3)
    wcss.append(kmeans.inertia_)

# Plotting the results onto a line graph, allowing us to observe the 'The
elbow'

plt.plot(range(1, 20), wcss)
plt.title("The elbow method for optimal k")
```

```
plt.xlabel("Number of clusters")
plt.ylabel("Error")
plt.show()
```

```
C:\ProgramData\Anaconda3\envs\tensorflow\lib\site-
packages\sklearn\cluster\_kmeans.py:1038: UserWarning: KMeans is known to
have a memory leak on Windows with MKL, when there are less chunks than
available threads. You can avoid it by setting the environment variable
OMP_NUM_THREADS=1.
  warnings.warn(
```

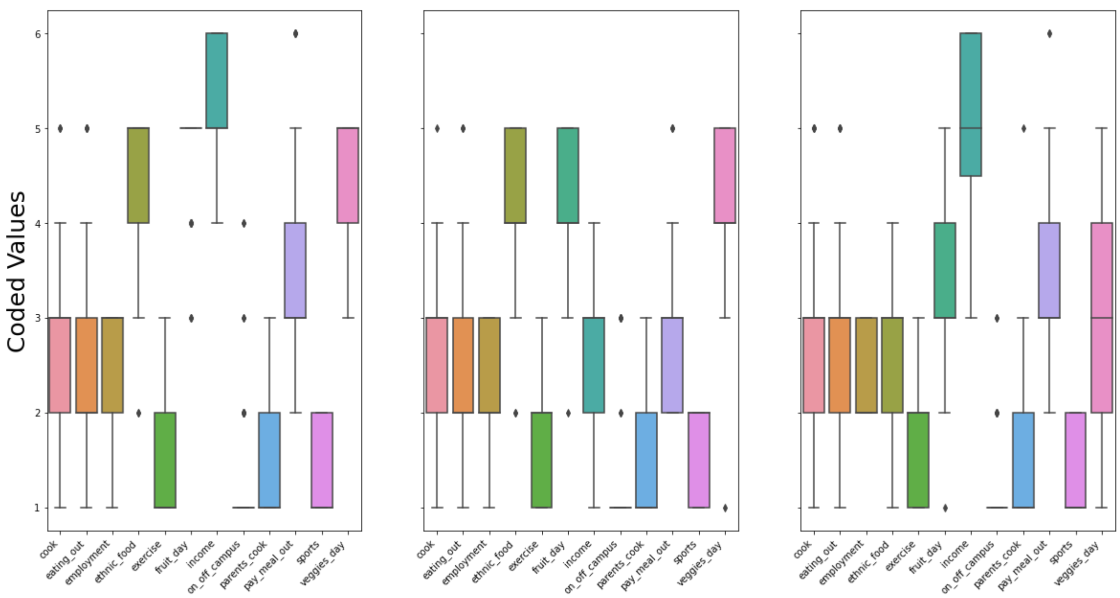


the elbow can be seen some what around the k value 3

```
# set number of clusters
kclusters = 3

# run k-means clustering
kmeans = KMeans(n_clusters=kclusters, random_state=0).fit(df3)
df3['Cluster']=kmeans.labels_

fig, axes = plt.subplots(1,kclusters, figsize=(20, 10), sharey=True)
axes[0].set_ylabel('Coded Values', fontsize=25)

for k in range(kclusters):
     plt.sca(axes[k])
     plt.xticks(rotation=45,ha='right')
     sns.boxplot(data = df3[df3['Cluster'] == k].drop('Cluster',1),
```

```
ax=axes[k])
plt.savefig("kmeans.png")
plt.show()
```

```
C:\Users\ROHITS~1\AppData\Local\Temp/ipykernel_13588/2049181312.py:7:
FutureWarning: In a future version of pandas all arguments of DataFrame.drop
except for the argument 'labels' will be keyword-only
  sns.boxplot(data = df3[df3['Cluster'] == k].drop('Cluster',1), ax=axes[k])
C:\Users\ROHITS~1\AppData\Local\Temp/ipykernel_13588/2049181312.py:7:
FutureWarning: In a future version of pandas all arguments of DataFrame.drop
except for the argument 'labels' will be keyword-only
  sns.boxplot(data = df3[df3['Cluster'] == k].drop('Cluster',1), ax=axes[k])
C:\Users\ROHITS~1\AppData\Local\Temp/ipykernel_13588/2049181312.py:7:
FutureWarning: In a future version of pandas all arguments of DataFrame.drop
except for the argument 'labels' will be keyword-only
  sns.boxplot(data = df3[df3['Cluster'] == k].drop('Cluster',1), ax=axes[k])
```



People with less income

1. people with less income(according to the data) tend to eat out less.
2. People with less income prefer to cook at home are or are more likely to cook at home.
3. People with less income eat out less often and want to buy grocries for home cook.

People with more income

1. As the income people increases(according to the data) their preference for ethnic food increases.
2. It is the same with food, they want to eat more fruits.

3. preference of veggies of veggies is also increased.

In conclusion the the people with less income need more shops and groceries around their place to buy veggies for home cook.

People with more need prefer to have ethnic restaurant around them, and shop to buy fruits

The number of clusters choose is 3 because it seems that data is nicely clustered around, income

Any less than that the visualization is messy

And more than that the visualization is also messy

```
search_query = 'PG Apartments' #Search for residential locations 28.364628,
77.539879
radius = 30000 #Set the radius to 30 kilometres due to remote college
location
latitude=28.364628 #College location(Galgotia's University)
longitude=77.539879
college = 'Galgotias University'

CLIENT_ID = 'S0KV55VKN1SDRR2B2Z1TIMA33Y230VSBQSFTQ1ASVJEUGSME'#
CLIENT_SECRET = 'L3Y3A3KCDNGSTD32Y2XTQQ0GEVU2WVYEFXPI1RPRU2PRJFTT' #
VERSION = '20210604'
LIMIT = 200

url =
f'https://api.foursquare.com/v2/venues/search?client_id={CLIENT_ID}&client_se
cret={CLIENT_SECRET}&ll={latitude},{longitude}&v={VERSION}&query={search_quer
y}&radius={radius}&limit={LIMIT}'

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

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

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

C:\Users\ROHITS~1\AppData\Local\Temp/ipykernel_13588/4005429704.py:5:
FutureWarning: pandas.io.json.json_normalize is deprecated, use
pandas.json_normalize instead
  dataframe = json_normalize(venues)
```

```
                         id                   name  \
0  4ec168ae30f82a2e13d52a40  Black Gold Apartments
1  535f3271498ee48607aea91a              PGDM Hall
2  4c80bcc7d34ca143a8ab1b80        Icon Apartments
3  51e4f94a498e834f69f32da2            Abhishek PG
4  57d1c577498efd6248631caf    Sai Park Apartments

                                          categories    referralId  hasPerk
\
0  [{'id': '4d954b06a243a5684965b473', 'name': 'R...  v-1638565310    False
1  [{'id': '4bf58dd8d48988d1a0941735', 'name': 'C...  v-1638565310    False
2                                                 []  v-1638565310    False
3  [{'id': '4d954b06a243a5684965b473', 'name': 'R...  v-1638565310    False
4  [{'id': '4d954b06a243a5684965b473', 'name': 'R...  v-1638565310    False

   location.lat  location.lng  \
0     28.455142     77.514105
1     28.470884     77.482033
2     28.461677     77.515641
3     28.541406     77.335435
4     28.412214     77.336770

                            location.labeledLatLngs  location.distance  \
0  [{'label': 'display', 'lat': 28.45514197248049...              10387
1  [{'label': 'display', 'lat': 28.47088432312011...              13114
2  [{'label': 'display', 'lat': 28.461677, 'lng':...              11060
3  [{'label': 'display', 'lat': 28.54140570515523...              28064
4  [{'label': 'display', 'lat': 28.412214, 'lng':...              20584

  location.cc location.country  \
0          IN            India
1          IN            India
2          IN            India
3          IN            India
4          IN            India

                          location.formattedAddress location.address  \
0                                         [India]              NaN
1                                         [India]              NaN
2                                         [India]              NaN
3  [Sector 126 (Opposite Amity Gate 2), Noida 201...       Sector 126
4                      [Farīdābād, Haryāna, India]              NaN

    location.crossStreet location.postalCode location.city location.state
0                    NaN                NaN           NaN            NaN
1                    NaN                NaN           NaN            NaN
2                    NaN                NaN           NaN            NaN
3  Opposite Amity Gate 2             201304         Noida  Uttar Pradesh
4                    NaN                NaN     Farīdābād        Haryāna
```

```
filtered_columns = ['name', 'categories'] + [col for col in dataframe.columns
if col.startswith('location.')] + ['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]
dataframe_filtered.drop([0, 35, 37, 17, 23, 26,27, 34, 28,
40],axis=0,inplace=True) #remove some unwanted locations like hotels
dataframe_filtered.drop(['cc','country','state','city'],axis=1,inplace=True)
#no need for those columns as we know we're in Greater Noida,IN
dataframe_filtered.head()
```

```
                 name                                 categories        lat
\
1           PGDM Hall                          College Classroom  28.470884
2     Icon Apartments                                       None  28.461677
3        Abhishek PG  Residential Building (Apartment / Condo)  28.541406
4  Sai Park Apartments  Residential Building (Apartment / Condo)  28.412214
5     Shanti Niwas PG  Residential Building (Apartment / Condo)  28.542465

         lng                                      labeledLatLngs  distance  \
1  77.482033  [{'label': 'display', 'lat': 28.47088432312011...     13114
2  77.515641  [{'label': 'display', 'lat': 28.461677, 'lng':...     11060
3  77.335435  [{'label': 'display', 'lat': 28.54140570515523...     28064
4  77.336770  [{'label': 'display', 'lat': 28.412214, 'lng':...     20584
5  77.338409  [{'label': 'display', 'lat': 28.54246520996093...     27941

                                   formattedAddress     address  \
1                                          [India]         NaN
2                                          [India]         NaN
3  [Sector 126 (Opposite Amity Gate 2), Noida 201...  Sector 126
4                       [Farīdābād, Haryāna, India]         NaN
```

```
5  [Sector 126 (Raipur), Noida 201304, Uttar Prad...  Sector 126

             crossStreet postalCode                        id
1                    NaN        NaN  535f3271498ee48607aea91a
2                    NaN        NaN  4c80bcc7d34ca143a8ab1b80
3  Opposite Amity Gate 2     201304  51e4f94a498e834f69f32da2
4                    NaN        NaN  57d1c577498efd6248631caf
5                 Raipur     201304  50f28cb0e4b0d13e7cb9e53f
```

```
#define coordinates of the college

map_Galgotia=folium.Map(location=[28.364628,77.539879],zoom_start=11)
# Marking College on the Map with red color.
folium.Marker([28.364628, 77.539879], popup=college, tooltip="Galgoitas
University").add_to(map_Galgotia)
# instantiate a feature group for the incidents in the dataframe
locations = folium.map.FeatureGroup()

latitudes = list(dataframe_filtered.lat)
longitudes = list( dataframe_filtered.lng)
names = list(dataframe_filtered.name)
distance = list(dataframe_filtered.distance.apply(lambda x : x/1000)) #
extracting distance from the data

for lat, lng, name, dist in zip(latitudes, longitudes, names, distance):
    # Note: the numbers are distance in Kilometers
    color = 'green'
    d = float(dist)
    if d < 10:
        color = 'pink'
    elif d >= 10 and d < 20:
        color = 'green'
    elif d >= 20 and d <= 30:
        color = 'orange'
    elif d > 30 and d <= 35:
        color = 'red'
    else:
        color = 'darkred'

    folium.Marker([lat, lng], popup=f"{name}, {dist}
KM",icon=folium.Icon(color=color), tooltip=name).add_to(map_Galgotia)

# add incidents to map
map_Galgotia.add_child(locations)

# add incidents to map
map_Galgotia.add_child(locations)
```

```
map_Galgotia
```

```
<folium.folium.Map at 0x21dc3811460>
```

1. blue marker is my college
2. green are the location between 10 and 20 kilomter
3. orange are the distance between 20 and 30 kilometer
4. red are the distance between 30 to 35 km
5. darkred are the distance greater than 35(not at all feasable and no location given)
6. pink could have the distance below 10 kilometer but data could not show that, but there are two location slightly greater than 10(green).

Chapter 7: Conclusion and Future Scope

There was no location around university, within 10 KM, and only two locations between 10 and 11 KM, that too are apartments.

The location data around Galgotias University is not very good, thus there might be some locations that didn't reflect in the above map

1. Students can look for apartments **around**, 'Knowledge Park' Area.
2. For better analysis need more data around the locations.
3. As we cross the 20 KM mark, we start to see some PGs and more apartments, there are pgs. **around** Amity university
4. You would need a transport facility to drop and pick you up from the university.
5. PGs are more clustered around Noida, but they are at a distance of more than 25 KM
6. There is one more university around Galgotias University, so business owners should think about building something there, that accommodates the students there.
7. The location each have at least 2 restaurants, 2 grocery stores and 2 markets
8. It would be better to choose according to income, the distance from university is a major factor.

Future Scope

This analysis can be used by business owner to build infrastructure around the discussed location, it will be going to be profitable as so many students are finding a place to stay.

This project can be scaled to cities and states to get more analysis around the habits of how people choose a place to stay, and the government act accordingly in-order to build more infrastructure.