Submitted by: -Piyush Gupta

Week 1 Deliverables:

Problem Statement: You run a company XYZ which operates in the Real State business, your company *(via its app/web application)* provides property recommendations to the customers (for buying/rental) who are going/coming abroad to settle or for long/short term vacation. Based on the customer profile/preferences, you recommend them the ideal or the best suited neighborhood in their choice (if any) of city.

In simple terms, customer from a **[Country/City/Neighborhood]** *A* wants to go to

[Country/City/Neighborhood] *B*, where B is supposed to be his/her the best possible option out there.

Now there are two scenarios:

1. The customer is *extremely well adjusted* to kind of surrounding she had at her old place **A** and would like to select a neighborhood **B** in the new city which closely matches **A** in terms of the great amenities and the venues in her neighborhood **A**
2. The customer is *not that satisfied* with her current neighborhood **A** and would like to select the new location **B** which matches some or all of her preferences.

   Note 1. *Both the above scenarios are essentially the same, as we would have a certain set of preferences in any case which we would like to match to the new location B, the only difference is that in the first case the preference list can be generated using the location data but in the second case the preferences would have to be fed manually into the system. (selection of options from a list of venue items)*

   Note 2. *For sake of simplicity in this project example we would take scenario 1 where customer is happy with her present location.*

SAMPLE CASE: A person P who lives in Toronto City <Address: > wants to move to New York City, and wants the best matching neighborhood as per his Toronto city address. He also wants the new place to be suitable as per his financial position. *(the affordability factor)*

APPROACH:

- **Getting the location profile:** First we will make the initial location profile of his Toronto address **(A)** using foursquare API, here we will make the
  GET https://api.foursquare.com/v2/venues/search or
  GET https://api.foursquare.com/v2/venues/explore call whichever suitable for the need.
- Now we have the list of venues at location A, then we will take inputs from the user to select the venues which cater to her needs the most. *(refining the venue profile)*. We will also ask the user to provide the explored RADIUS parameter to get as many relevant data points as possible.

- **Financial Parameters:** At this point we have the venue profile, since we have to recommend the new location based on her affordability factor as well; next we would ask for the user's **annual salary** and the expenses occurring per month.

  ➢ **Data needed:** Salary, Present Rent, Approx. present fixed expenses (groceries/entertainment/leisure etc.), this data would be used to refine the selection of recommended neighborhoods based on cost of living adjustments. We can also ask for something like the **stretch in cost** factor which the user is willing to pay beyond her previous expenses. This value should default to 1 implying no substantial addition of cost should be there in the new location. For example: present total fixed expenses = **100 $**, stretch in cost factor = **20%**, new maximum total fixed expense can be up to **120$**.

  ➢ **Additional Data Sets required for related analysis:**

    To make use of the financial parameters we would need,

    1. Neighborhood level rent prices(mean/median)
    2. Cost of living data
    3. Wages data

VENUE_PROFILE ➕ **FINANCIAL_PARAMETRS** 🟰 **CUST_PROFILE**

- **Segmentation/Clustering of New York city:** He we will use the code based generated in the case of New York City clustering (done in the lab session). After the clustering part*(KMeans)* is done we will predict the cluster label of our venue profile *(from Toronto address)*. Now we have a cluster made up of similar location as her previous venue_profile, now we can leverage the financial_params to find the most suitable neighborhood using KNN algorithm.