DATA REQUIREMENTS AND METHODOLOGY

IBM Data Science Certification Capstone Project

1. Data Acquisition and Literature Review

In order to solve the problem at handle a large amount of reference, geospatial and pricing data will be required. For this I will make use of several tools, libraries and APIs, as follows:

1. Geopy Geocode Nominatim [4]: Geopy is a geospatial client for Python and other web services. Geopy makes it easy for Python developers to locate the coordinates of addresses, cities, and landmarks across the globe and includes classes for the OpenStreetMap Nominatim, Google Geocoding API (V3), and many other geocoding services

2. Foursquare Places API [5]: Foursquare is the most trusted, independent location data platform for understanding how people move through the real world. Foursquare currently has over 60 million registered users sharing locations, venues, pictures, ratings etc around the globe. The Foursquare Places API offers real-time access to Foursquare's global database of rich venue data and user content to power location-based experiences. With a Personal developer account one can commit 99500 standard calls and 50 complex calls to the Foursquare API per day, with endpoints ranging from Venue names to user reviews.

3. PSRA - Dublin Residential Property Price Register [6]: This is openly available data published by the Property Services Regulatory Authority in Ireland. It includes Date of Sale, Price and Address of all residential properties purchased in Ireland since the 1st January 2010. In its present form it does not possess geographical coordinates, so Geopy will need to be used to enrich the data further. Note that Geopy is notoriously unstable and using Geocode on thousands of addresses (most of them with peculiar Irish names and areas) takes an incredibly long time and will also produce many null values. As such, I have included the code in my Juptyer Notebook, but for simplicity have saved the geographical coordinates to a csv file obtained from The Irish Property Price Register – Geocoded to Small Areas [7].

4. BeautifulSoup [8]: BeautifulSoup is a Python library that makes it easy to scrape information from web pages. It allows an HTML script to be parsed, iterated, searched, and modified.

2. Methodology & Exploratory Data Analysis

Given that there are three specific criteria to fulfil in order to answer the problem statement, the following methodology and analysis was used for each one:

3.1 Overview of Methodology

3.1.1 Area Clustering (finding similar areas in Dublin to Morningside, Johannesburg):

K-Means clustering of all areas in Dublin and Morningside, Johannesburg will be done in order to visually identify places in Dublin that are similar to Morningside. The K-Means clustering machine learning algorithm was chosen for this as it allows the greatest flexibility in selecting and viewing various values of K (number of clusters), as well as the fact that it is fast, robust, and gives reliable results for data sets that are distinct and well separated from each other. This suits the dataset well. The overview of the methodology to achieve this is:

- The geographical coordinates of the various Dublin postal codes will be obtained using Geopy Nominatim Geocoder. I will then manually add my desired South African address in Morningside, Johannesburg to the list of addresses so that its geographical locations are added as well.
- Next, I will calculate the distance from the centre of Dublin City (near to where I will be working) to all the Postal Code Areas using a Great Circle (or Hoversine) formula that I will define as a function and remove all areas that are further than 10km's away (except for Morningside, Johannesburg as I want to compare this later).
- Once complete, the Foursquare Places API will be used to scour the surrounding area within a 3km radius of the geographic locations of each Postal Code obtained from Geopy, and the various venues will be collated.
- The OneHotEncoding technique will then be used to create dummy variables of all venues and the frequency of venues within the area will then be calculated and sorted from most to least frequent.
- The K-Means Clustering Machine Learning algorithm will then be utilised to compare all areas to each other and create clusters of areas that are most alike. These results will be displayed on a Folium map. This will indicate which areas of the 22 areas in Dublin are most like Morningside, Johannesburg and are less than or equal to 10km from the City centre.

3.1.2 Examining Dublin's Housing Prices by Area

The K-Means clustering algorithm above, if done correctly, will produce various clusters showing the different areas of Dublin which are similar to Morningside, Johannesburg. However since I am looking for one definitive result I will need to dig deeper into other factors, such as housing prices, to narrow it down. My limit for buying a house is €500,000, and cheaper is certainly preferable. This section will attempt to a) understand the layout of Dublin and where the various low-cost and expensive areas are, and then b) rank the various areas by their average house prices to focus the decision:

- The Dublin Residential Property Price Register data will be read into Python and cleaned to remove blanks, certain unused columns and excessively high or low prices (outliers).
- The data will the be enriched with geographical coordinates so that it can be mapped (this can be done with the Geopy Geocoder but as per section 2 part 3 the coordinates have been saved to a csv file to save time so that it can be read in directly).
- Data analysis of the price will be performed in several ways including placing each house into a price 'bucket' and colour coding it so that, when mapped onto a Folium map, it is possible to see the prices of different areas and how they change.
- Finally, the average price of the houses within each area will be obtained by grouping the 3000 houses by area and then aggregating them. Once the average price is obtained the areas will be ranked from cheapest to most expensive.

3.1.3 Finding the best Vibe – or in this case, the top 10 Irish Pubs

They say that the best thing about Ireland, other than the Galway Girls and the Leprechauns, are the Irish Pubs. They are a large part of Irish culture and entertainment. As such, although this is a lower priority than sections 3.1 and 3.2, the aim is to find out a) where the top 5 Pub areas are in Dublin by looking at the average number of 'Likes' for various pubs scraped using the Foursquare API and b) find what the names are of the top 10 pubs by number of 'Likes' (as well as where to find them. Using 'Likes' is not necessarily the most accurate measure as many people don't leave 'Likes' for a venue, but it will be used as a fun exploration of the city.

- Once again Foursquare Places API will be utilised to search the areas suggested as being most like Morningside, Johannesburg in section 3.1 for venues with the keyword "Pub" within a 3km radius.
- The data will be further cleaned to ensure that irrelevant venues (such as "Public library", or 'Garden Bar') are removed by filtering on category. There may also be overlaps between areas, so duplicates will be removed, keeping the area name that is closest to the pub.
- The number of 'Likes' for each Pub will be determined. These will then be averaged and grouped by area and then ranked from most to least 'Likes'.
- A list of all pubs will be created, sorted from highest to lowest, in order to obtain a top 10 list. Their locations on a Folim map will be plotted in order to better understand where in the city they are.
-

3.4 Deciding Where to Live

The initial investigation starts with 24 potential areas in Dublin in which to live. Utilising the 10Km rule, and the clustering information from section 3.1 as a primary guide, the goal is to first narrow this number down. Second, using the housing data ranking and a budget of €500 000 at maximum (preferably cheaper) the choices will hopefully further be narrowed. Lastly, the pub rankings will hopefully reveal the areas with a great vibe to allow for the selection a specific area (and of course find the best pub in town!)