PINPOINT PERFECT PARISH (Porto City)

Ivan Kisialiou

28.04.2019

2. DATA

In order to do clustering over Porto City districts (Parishes), I'll consider venues within each parish together with distances from the parish centre to the City Centre ("Remoteness"), to the sea shore, to the Porto University (FEUP), and to the airport.

The customer's preferences will be included to a model as weights.

2.1. Parishes of Porto

First, we need names and coordinates of geographical areas in Porto. Portuguese administrative division is quite complicated and includes 18 districts (https://en.wikipedia.org/wiki/Administrative_divisions_of_Portugal).

Porto District comprises 18 Municipalities, and each Municipalities has several Parishes ("Freguesias") in it. There is no list of Parishes in Porto Agglomeration, so we can select them by their remoteness. The complete list of Portuguese Parishes can be found here:

https://pt.wikipedia.org/wiki/Lista_de_freguesias_de_Portugal

I carried out parsing of this page and selected Parishes in Municipalities of Porto District.

I used *OpenStreetMap Nominatim* (https://nominatim.openstreetmap.org/) for geocoding.

Cleaning: dropped Parishes that are too far (>10 km) from the City Centre; split several joint Parishes (e.g. "Aldoar, Foz do Douro e Nevogilde") to obtain more points om map; moved one geo-point ("Matosinhos") to its true position.

The resulting dataframe and the map with Parishes are shown in Fig.1 and Fig.2.

	Municipality	Parish	Latitude	Longitude	Remoteness_km
0	Gondomar	Baguim do Monte (Rio Tinto)	41.187473	-8.537759	7.4
1	Gondomar	Fânzeres	41.171488	-8.531211	7.1
2	Gondomar	São Pedro da Cova	41.153457	-8.506091	8.8

Figure 1. Dataframe with geo-data for Porto City Parishes

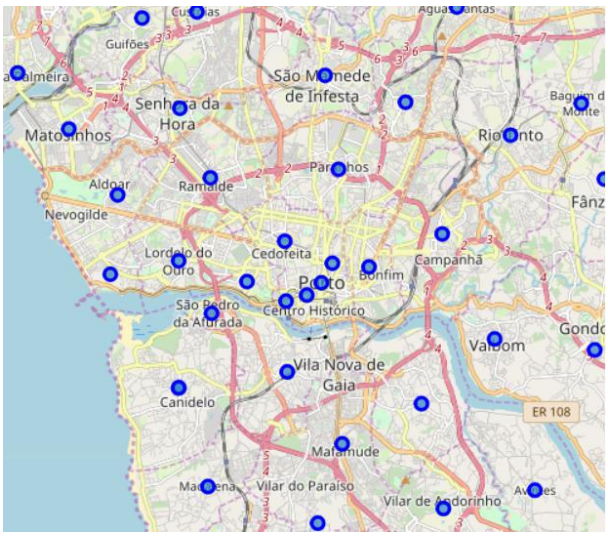

Figure 2. A map with the centres of Porto Parishes on it

2.2. Venues

I applied Foursquare API to obtain list of maximum 100 most popular venues for each Parish. The request form is shown below.

```
# create the API request URL
url = 'https://api.foursquare.com/v2/venues/explore?&client_id={}&client_secret={}&v={}&ll={},{}&radius={}&limit={}'
.format(
    fs.id, # user's ID
    fs.secret, # user's Secret
    fs.version, # API version
    lat, # latitude
    lng, # longitude
    radius, # radius to explore
    limit)  # number of venues to return
```

Figure 3. Foursquare API request for fetching the most popular venues within the certain radius from given geo-point

The radius parameter for request with endpoint “explore” was set to the half of average distance between two neighbouring Parishes in Porto. To do that I had to write the code calculating this parameter for a set of geo-points.

The resulting dataframe is as follows:

	Municipality	Parish	Parish Latitude	Parish Longitude	Venue	Venue Latitude	Venue Longitude	Venue Category
0	Gondomar	Baguim do Monte (Rio Tinto)	41.187473	-8.537759	O "Zé Pacheco"	41.188296	-8.539716	Food
1	Gondomar	Baguim do Monte (Rio Tinto)	41.187473	-8.537759	Lidl	41.182358	-8.537666	Grocery Store
2	Gondomar	Baguim do Monte (Rio Tinto)	41.187473	-8.537759	Tentação das Bifanas	41.184127	-8.539149	Restaurant

Figure 4. Dataframe with the data about venues in Porto City Parishes

2.3. Distances

Besides “Remoteness” we are going to consider distances to the Airport, to the seashore, and to the University, as our sample customer is a student of The Porto University (FEUP). Distances to the Airport and to the University can be calculated

in the same way as “Remoteness” – with the help of the method *.distance* from geopy library. The calculation of the distance to the seashore (line shown in Fig. 5) requires the knowledge of cross-track distance concept.

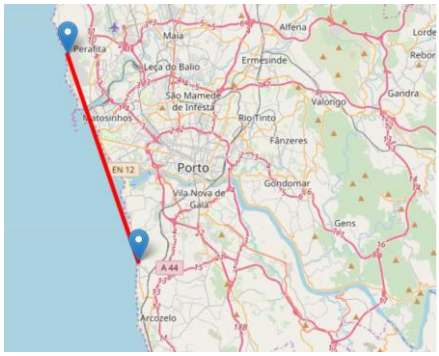

Figure 5. The model line representing the seashore for distance calculation

The cross-track distance can be analytically calculated with the help of mathematical formulas, but fortunately there is n-vector library for Python.

2.4. Preferences

We can formalize the customer’s preferences about segmentation criteria by asking whether the given feature is important for their choice, or not. Then the answers are transformed into the weight vector, which we apply to the features. The features are described in Feature Engineering section.