# Helping tourists go through the city

Enrique Olmedo

# . Introduction

## Problem

Tourism is an activity that moves people and dollars all around the world. Every tourist wants to enjoy their stay in a foreign city or country. However, multiple problems arise when visiting other places such us concerns about security, know which places to visit and making every minute and second count.

This work focuses on the last two problems mentioned above, namely, it gives a guide on which places to visit according to their popularity and optimizes the traveler’s route so that she doesn’t spend any minute longer than she needs to moving from one place to another.

The city chosen to solve this problem is my hometown, Santiago, Chile.

## Interest

The people who would be interested in this kind of solution are people who like to travel and do it often. Most importantly, people who don’t want to miss any good place and make every minute count while on vacation, such us myself.

# . Data acquisition and cleaning

## Data sources

Data and ratings from the venues in Santiago are taken from the Fourquare API. Data from the city’s ‘comunas’ is taken from shapefiles maps found at the [National Congress Library Site](https://www.bcn.cl/siit/mapas_vectoriales) and handled using the geopandas library.

## Data cleaning

There were several problems with the data. The most important one is that the free version of Foursquare API returns 100 entries maximum per request. However, this was solved using the comunas in which the city is divided, so I was able to return 100 venues per comuna which is enough for the scope of this work. This gave me another problem: requests made to Foursquare API are made assuming regular shapes (a circle), but comunas don’t have regular shapes so some venues where assigned to comunas where they don’t belong. I decided to partially correct this by assigning the corresponding comuna if the venue had the comuna’s name in the venue’s name.

The last problem is that in Chile, spanish is spoken and Spanish has some special characters such as tildes and ‘ñ’. So some venue’s and comuna’s names are duplicated. For example, you may find the comuna ‘Peñalolén’ as ‘Peñalolen’, ‘Penalolén’ or ‘Penalolen’. I fixed this issue by manually correcting every misspelled name I could find.

## Feature selection

The features that were selected from the Fourquare API were: Venue Id, Venue name, Venue Latitude, Venue Longitude, Venue Category and Venue Ratings.

Features that were selected from the information on the map are: Comuna Centroid Latitude, Comuna Centroid Longitude and Comuna Name.

The only feature that was calculated was the Distance to Comuna Centroid.

## Exploratory Data Analysis.

### Comunas with most venues

The 11 comunas with the highest number of venues are summed up in Table 1

|  |  |
| --- | --- |
| **Comuna** | **Number of venues** |
| **Santiago** | 113 |
| **Ñuñoa** | 99 |
| **Providencia** | 99 |
| **Maipú** | 98 |
| **Puente Alto** | 96 |
| **Vitacura** | 96 |
| **La Reina** | 93 |
| **San Bernardo** | 81 |
| **La Florida** | 80 |
| **San Miguel** | 78 |
| **Las Condes** | 74 |

Table 1. Comunas with highest number of venues

### Most Popular Comunas

Table 2 sums up the 10 comunas with highest ratings number

|  |  |
| --- | --- |
| **Comuna** | **Ratings number** |
| **Providencia** | 20272 |
| **Santiago** | 18429 |
| **Ñuñoa** | 12700 |
| **Pudahuel** | 9340 |
| **Vitacura** | 8961 |
| **Las Condes** | 5471 |
| **San José de Maipo** | 3767 |
| **La Florida** | 3497 |
| **La Reina** | 3490 |
| **Lo Barnechea** | 3419 |

Table 2. Comunas with highest ratings number.

Table 3 shows the most common venues for the most popular comunas.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Comuna** | **1st Most Common Venue** | **2nd Most Common Venue** | **3rd Most Common Venue** | **4th Most Common Venue** | **5th Most Common Venue** |
| **Providencia** | Sandwich Place | Restaurant | Bar | Pizza Place | French Restaurant |
| **Santiago** | Coffee Shop | Sandwich Place | Pizza Place | Park | Café |
| **Ñuñoa** | Bakery | Italian Restaurant | Restaurant | Café | Bar |
| **Pudahuel** | Airport Lounge | Coffee Shop | Hotel | Sushi Restaurant | Soccer Field |
| **Vitacura** | Café | Sushi Restaurant | Gym / Fitness Center | Italian Restaurant | Bakery |
| **Las Condes** | Bakery | Coffee Shop | Supermarket | Fast Food Restaurant | Multiplex |
| **San José de Maipo** | Scenic Lookout | Mountain | Restaurant | Hotel | Campground |
| **La Florida** | Sushi Restaurant | Chinese Restaurant | Pizza Place | Gym | Soccer Stadium |
| **La Reina** | Restaurant | Sushi Restaurant | Soccer Field | Chinese Restaurant | Pizza Place |
| **Lo Barnechea** | Ski Area | Hotel | Mountain | Restaurant | Trail |

Table 3. Most common venues for most popular comunas

# Optimization Problem

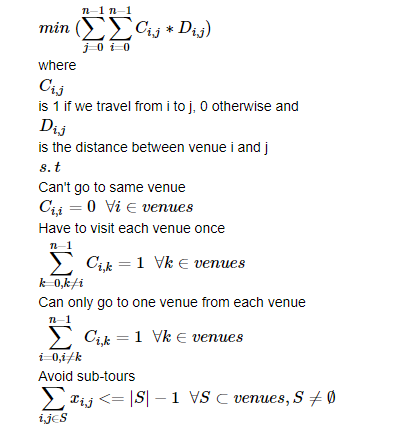
## Data preparation

To optimize our travelling day, first we have to input the categories of places that we would like to visit. Then, the program gets the most popular venue for each category and it’s latitude and longitude.

Finally, we get a distance matrix for all of the venues mentioned above.

## Travelling Salesman Problem

For route optimization, I used linear optimization methods and based my problem on the travelling salesman problem. So we have to solve the following problem using the library pulp.



# Results

For this example, I used the following venue categories:

* Stadium
* Sushi Restaurant
* Park
* Art Museum

The program returned the following venues for each category:

* Movistar Arena
* Niu Sushi
* Parque Padre Hurtado
* Palacio La Moneda

The results for routes for our problem are the following:

(0, 0) 0.0

(0, 1) 0.0

(0, 2) 0.0

(0, 3) 1.0

(1, 0) 0.0

(1, 1) 0.0

(1, 2) 1.0

(1, 3) 0.0

(2, 0) 1.0

(2, 1) 0.0

(2, 2) 0.0

(2, 3) 0.0

(3, 0) 0.0

(3, 1) 1.0

(3, 2) 0.0

(3, 3) 0.0

And our output to the user is:

Start in Parque Padre Hurtado in Padre Hurtado

Then go to Niu Sushi in Vitacura

Then go to Centro Cultural Palacio La Moneda in Santiago

Then go to Movistar Arena in Santiago

# Discussion

This approach is a good first approach to optimize routes and ratings for tourists. However, a few setbacks come to mind with this solution. First, the distance is taken using a latitude and longitude projection so it has errors. Second, it doesn’t take into account the grid of the streets nor traffic. Third, it doesn’t take into account time (for example, assign the restaurant venue to visit at lunch time). A better solution to this would be to use the Google API alongside the Foursquare API, that approach could help us obtain traffic information and the streets grid at real time.

Further work could be done by making personalized recommendations to the user by creating a collaborative filter recommendation engine, using ratings of those users that fit the user’s persona.

# Conclusions

In this work, I have established a first approach to optimize routes and ratings in places to visit for tourists and travelers. This problem was approached by using ratings and venue information from the Foursquare API which proved to be very useful for this type of problem. Finally, I optimized distances and routes using optimization algorithms and establishing an optimization problem based on the travelling salesman problem.