

Determining Optimal Food Truck Locations

Evaluating Neighborhoods in Austin, Texas

Introduction

The audience for this data science problem will be owners and operators of traveling food trucks. As food truck operators enter a new city and a new market, a persistent question will be: where to park the foodtruck for maximum revenue. We will attempt to use data science and data from Foursquare to allow foodtruck operators to make a data-informed decision about where to park their food truck.

In this project we will determine the optimal Austin neighborhood to park a new food truck in. We will understand each Austin neighborhood's proximity to popular nightlife spots, as well as how saturated the food truck market is in the given neighborhood.

Data

In order to solve this problem we will firstly need data on Austin neighborhoods. As no dataset readily exists for Austin, Texas we'll have to scrape Wikipedia for neighborhood information. [This Wikipedia page](#) contains a list of Austin neighborhoods. Clicking a neighborhood shows that most pages have a *Coordinates* section at the top which links to a [geohack page](#).

Food trucks perform best in areas with a lot of nightlife spots. So we'll want to measure how many bars and nightlife spots are close to each neighborhood. We'll also want to measure how many foodtrucks are already in certain neighborhoods, to find neighborhoods that are less competitive. All of this data can be obtained through the [Foursquare API](#).

Here is an example of using the Foursquare API in Python to retrieve nightlife spots near Austin, Texas.

```
use-foursquare.py

#!/usr/bin/python
import pandas as pd
import requests

def venues(latitude, longitude, category):
    """ Returns venues within a 1 km radius of a location """
    url = 'https://api.foursquare.com/v2/venues/explore?client_id={}' \
        '&client_secret={}&ll={},{}&v=20201120&categoryId={}' \
        '&radius=1000&limit=50'.format(client_id, client_secret, \
                                       latitude, longitude, category)
    results = requests.get(url).json()
    if results['meta']['code'] == '429':
        raise ValueError('Foursquare_Quota_Exceeded')
    return pd.json_normalize(results['response']['groups'][0]['items'])

df_n1 = venues(30.3076863, -97.8934851, '4d4b7105d754a06376d81259')
```

And here is the resulting information:

Venue	Latitude	Longitude
The Common Interest	30.366192	-97.729144
Slick Willie's Family Pool	30.366677	-97.729049
Water 2 Wine	30.362011	-97.742086
Nosh & Bevvv	30.365414	-97.728802
Buddy's	30.368978	-97.727205