

## README: San Fransico Real Estate Recommendation App

**MUST DOS:** Please install geopy using pip install geopy, or any other alternative methods.

- Running the program is simple, run the python script via any code editor, and input any address. There may be bugs with the dataframe, where the address is unavailable, you'll be prompted to submit another address. You should use the addresses in the same outputs, as they show the expected outputs.

**Tech Stack:** Python

### **Libraries/Frameworks used:**

- Pandas + Numpy (used to clean data)
  - o Pandas was used to load JSON file via San Fransico's API into a Dataframe
  - o Numpy used to do operations. (Havrsine, and math on longitutde and latitude.)
- Geopy
  - o Used to get the longitude and latitude given an address
- Matplotlib
  - o Used for the visuals for the recommendations (graphs)

### **Data Sources used:**

- I wanted to keep this project solely on the use of San Francisco data, as there was already too much information on this API, and some of which are slightly unreliable (i.e, look at the 'the\_geom' column, it says some of it is unreliable, but it was the only viable way to locate the buildings because the property\_location column is confusing).
- Used Zillow for non-programming reasons, such as finding houses that are currently on sale (could be an issue because these houses may be new constructions and as such, not in the database).

### **Challenge:**

The property dataset contains addresses in non-standard or inconsistent formats (e.g., "0000 0180 07TH ST0202"), making direct string matching unreliable or impossible.

### **Proposed Solution:**

To accurately identify the property referenced by a user-inputted address, I implemented a geospatial matching method:

1. **Geocode the input address:**

Using the geopy library, the user's provided address is converted into geographic coordinates (latitude and longitude).

2. **Find the closest property in the dataset:**

Since the dataset includes geospatial information (the `_geom` field with coordinates), I calculate the geographic distance (using the Haversine formula) between the input address coordinates and every property's coordinates in the dataset.

3. **Select the property with the smallest distance:**

The property whose coordinates are closest to the geocoded input address is assumed to be the correct match. This avoids unreliable string matching and leverages spatial proximity instead.

4. **Output the matched property details:**

For debugging and transparency, the matched property's information (address, characteristics) is displayed.

5. **Use this matched property as the reference** for finding comparable properties nearby based on features such as bedrooms, bathrooms, square footage, price, etc.

### **Comparisons**

To determine which properties are "comparable," I focused on key features that most strongly influence real estate value and buyer preferences. These features are:

- **Location (Latitude & Longitude):** Proximity is critical; properties close to each other tend to share neighborhood characteristics, school districts, and amenities. I limit comparable to within a reasonable distance (e.g., 2 miles).
- **Number of Bedrooms:** Buyers typically compare homes with similar bedroom counts to meet their space needs. However, there is not a big deduction if the building had a little.

- **Number of Bathrooms:** Bathrooms significantly affect livability and value, so similar bathroom counts are important. Same idea as bedrooms, a difference of 1 is not too bad, but once we hit the difference of 2, its heavily punished (say, a family 4 living in a 4-bedroom 3 washroom house, suddenly has only 1 washroom to share).
- **Square Footage:** Size of the living space directly influences price and usability.
- **Price per Square Foot:** This normalizes price by size, allowing fair comparison of value.
- **Overall, Price:** Helps avoid comparing drastically different market segments.

Each feature is weighted to calculate a **similarity score** that ranks how close a candidate property is to the target property. For example:

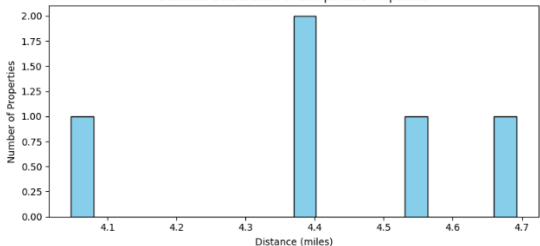
- Location proximity is heavily weighted since neighborhood effects matter a lot.
- Bedroom and bathroom counts are important but allowed some flexibility.
- Square footage and price per square foot help capture market value and property scale.

This multi-factor approach balances physical characteristics and market value to find truly comparable properties that buyers and appraisers would reasonably consider alternatives.

### Sample Outputs.

Found Comparable Properties: (keep in mind, the outputs MAY change because of ties. Pandas has its own way of solving tie breakers). I am aware some of the outputs are 0, this is purely because of the dataset having 0's randomly, and pruning these rows (properties) would've been bad because its already hard to find the specific property given an address, so removing more options would complicate the process.

Input: 30 Bird St, San Francisco, CA 94110	Input: 2066 Quesada Ave, San Francisco, CA 94124
Found Comparable Properties:	Found Comparable Properties:
Address: 0298 0290 07TH ST0000 Bedrooms: 0.0, Bathrooms: 4.0 SqFt: 13600.0, Price: \$3,231,990 Distance: 2.07 miles -----	Address: 0343 0301 LANGTON ST0000 Bedrooms: 0.0, Bathrooms: 0.0 SqFt: 6720.0, Price: \$907,748 Distance: 4.05 miles -----

<p>Address: 0000 0230 07TH ST0000  Bedrooms: 0.0, Bathrooms: 2.0  SqFt: 14230.0, Price: \$3,713,640  Distance: 2.07 miles</p> <p>-----</p> <p>Address: 0485 0475 06TH ST0000  Bedrooms: 0.0, Bathrooms: 3.0  SqFt: 14250.0, Price: \$2,822,524  Distance: 2.40 miles</p> <p>-----</p> <p>Address: 0000 0928 HARRISON ST0000  Bedrooms: 0.0, Bathrooms: 3.0  SqFt: 15150.0, Price: \$3,355,006  Distance: 2.54 miles</p> <p>-----</p> <p>Address: 0237 0235 9TH ST0000  Bedrooms: 0.0, Bathrooms: 3.0  SqFt: 8000.0, Price: \$2,263,540  Distance: 1.67 miles</p> <p>-----</p>	<p>Address: 0000 0933 HARRISON ST0000  Bedrooms: 0.0, Bathrooms: 0.0  SqFt: 6136.0, Price: \$732,241  Distance: 4.56 miles</p> <p>-----</p> <p>Address: 0000 0123 LANGTON ST0000  Bedrooms: 0.0, Bathrooms: 1.0  SqFt: 8000.0, Price: \$970,156  Distance: 4.39 miles</p> <p>-----</p> <p>Address: 0000 1174 FOLSOM ST0000  Bedrooms: 0.0, Bathrooms: 0.0  SqFt: 4220.0, Price: \$477,518  Distance: 4.38 miles</p> <p>-----</p> <p>Address: 1145 1149 MISSION ST0000  Bedrooms: 0.0, Bathrooms: 0.0  SqFt: 8500.0, Price: \$764,809  Distance: 4.69 miles</p> <p>-----f</p> <p>Distance Distribution of Comparable Properties</p>  <table border="1"> <caption>Distance Distribution of Comparable Properties</caption> <thead> <tr> <th>Distance (miles)</th> <th>Number of Properties</th> </tr> </thead> <tbody> <tr> <td>4.1</td> <td>1.00</td> </tr> <tr> <td>4.4</td> <td>2.00</td> </tr> <tr> <td>4.5</td> <td>1.00</td> </tr> <tr> <td>4.7</td> <td>1.00</td> </tr> </tbody> </table>	Distance (miles)	Number of Properties	4.1	1.00	4.4	2.00	4.5	1.00	4.7	1.00
Distance (miles)	Number of Properties										
4.1	1.00										
4.4	2.00										
4.5	1.00										
4.7	1.00										

