Project2

March 10, 2024

1 What are the impacts of property attributes on the the US Real Estate Market ?

1.1 Introduction

The objective of this research is to conduct a comprehensive analysis and build predictive models for key determinants influencing housing pricces. Factors contributing to a property's value are examined to seek uncover trends and patterns of US real estate market. The resulting models will provide a crucial observations and serve as a realiable resource for buyers, investors, and sellers. Those observations are insightful and help in facilitating informed decision-making in the dynamics on housing market.

1.1.1 Data Source

The dataset of housing prices was obtained from Kaggle, containing Real Estate listings in the US by State and zip code. It is collected from https://www.realtor.com/ - a real estate listing website operated by the News Corp subsidiary Move, Inc. and based in Santa Clara, California. The datase has 1 CSV file with 10 columns, each column represents a factor of the property. In combination with the real estate dataset, a second dataset was collected from the United States Census Bureau, specifically the 2020 Census Demographic Profile. This supplementary dataset contributes demographic insights and serves as a valuable complement to the real estate data.

1.1.2 Background

The research focuses on how a housing features (number of bedrooms, bathrooms, house sizes, and location) can have an impact on housing prices in the USA. Those factors are important features of a property and a key determinants of how a house is valued in the market. To unravel the relationship between the ethnic groups and housing prices, the paper employs analytical plots and distribution of those factors to determine how they correlate with the values of property.

Besides, different ethic groups across US states might potentially influence properties' prices in the USA. The focusing groups are White, Asian, and Black/African population. According to Journal of Urban Economics, racial and ethnic price differentiates in the housing market (Bayer et al., 2019). This additional exploration will make the approach more comprehensive in acknowledging the relationship of housing prices across the USA with other important factors.

1.2 Data Preprocessing

- Construcing the Summary Statistics Table

[ ]:
```python
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
import numpy as np
import warnings
import plotly.graph_objects as go

# Suppress RuntimeWarnings from NumPy
warnings.filterwarnings("ignore", category=RuntimeWarning)
# Loading the dataset
df = pd.read_csv('/Users/hanhhieudao/Desktop/ECO225/realtor-data.zip.csv')
df.head(5)
```

[ ]:
```
     status  bed  bath  acre_lot        city        state  zip_code  \
0  for_sale  3.0   2.0      0.12    Adjuntas  Puerto Rico     601.0
1  for_sale  4.0   2.0      0.08    Adjuntas  Puerto Rico     601.0
2  for_sale  2.0   1.0      0.15  Juana Diaz  Puerto Rico     795.0
3  for_sale  4.0   2.0      0.10       Ponce  Puerto Rico     731.0
4  for_sale  6.0   2.0      0.05    Mayaguez  Puerto Rico     680.0

   house_size prev_sold_date     price
0       920.0            NaN  105000.0
1      1527.0            NaN   80000.0
2       748.0            NaN   67000.0
3      1800.0            NaN  145000.0
4         NaN            NaN   65000.0
```

Some of the NA values were dropped since Matplotlib and Seaborn libraries in Python do not handle NA values well. Removing them from the data will better the visualization of the data without causing error in compiling codes.

[ ]:
```python
# Checking for null/missing values
df.isnull().sum()
```

[ ]:
```
status                 0
bed               216528
bath              194213
acre_lot          357467
city                 191
state                  0
zip_code             479
house_size        450112
prev_sold_date    686293
price                108
dtype: int64
```

The dataset exhibites a signficant amount of missing values, specifically in columns: bed, bath, acre_lot, city, zip_code, house_size, prev_sold_date, and price. Those missing values might raise potential biases in decriptive stastitics and potential impact on our analytical insights. A summary statistics table will be useful to give an overview of the distribution of available values with their central tendency and spread.

```
[ ]: summary = df[['bed', 'bath', 'acre_lot', 'price']].describe()
     summary.columns = ['Bedrooms', 'Bathrooms', 'House Size (sqft)','Price␣
      ↪(thousand USD)']
     table = summary.style.format({
         'Bedrooms': '{:.1f}',
         'Bathrooms': '{:.1f}',
         'House Size': '{:.1f}',
         'Price (thousand USD)': '${:,.0f}'
     }).set_caption('Summary Statistics')
     table
```

```
[ ]: <pandas.io.formats.style.Styler at 0x299eedb90>
```

The table provide some key insights of the dataset. In terms of number of bedrooms ('bed'), a diverse range is observed with average of 3 bedrooms with a standard deviation of 2, indicating variability around this mean. However, it has some outliers as the max number of bedroosm goes up to 123. Similarly, the bathroom counts have an avergae of 2 bathrooms per property, but there's also a property with 198 bathrooms, rasing some concerns about outliers. These findings underscore the necessity for meticulous outlier detection and critical assessment of the dataset's reliability in accurately representing properties across the state. In addition, outliers also indicate that larger property with high number of amentities are highly valuable in the market. Comparing to normal properties with average of 2-3 bathrooms and bedrooms, those large property demonstrate their uniqueness in the number of house rooms have extremely high values.

The distribution of living spaces might be right-screwked, with a tail extending towards larger sizes. This is due to the presence of outliers, and since 75% of the data is below 2500, it indicates that the majority of thoe houses have sizes on the lower end. The data implies that living space is a very important determinant to drive up the housing prices, rising up the property values significantly to millions of USA.

The average house price is $755,479 but the high standard deviation of $1,030,817 represents a wide range of prices of houses across the USA. This distribution suggests that there are various factors contribute to the price fluctuations. To address this spread, it's crucial to not only focus on housing properties, but also their other key factors such as neighborhood characteristics, socio-economic factors, and public amentities.

1.3 Variables Selections

1.3.1 Dependent variable (Y)

Housing price represents the monetary value of the property in the market. This is the target variable for pricing analysis of real estate across USA regarding to chosen indepedent variables.

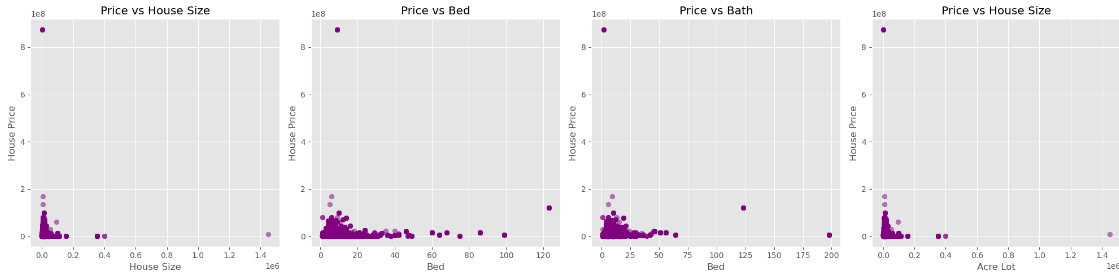
1.3.2 Independent variable (X)

1. Number of bedrooms 'bed': The number of bedrooms represents the aspect of residential properties and influence a house's market values. Besides, it also provides insights into diversity of personal preferences and different need of house buyers. For instance, they can use extra rooms in the house for guest rooms, home offices, etc. that reflect a very distinct buyers segments within real estate market.

2. Number of bathrooms 'bath': The number of bathrooms is a fundamental utility for homeowners and potential buyers. It contributes to the functional aspect of a property and also reflece the needs of inidviduals.

3. House size 'house_size': This matters because a bigger living space usually comes with more amentities and features, making the house more valuable. How the size of a house connected to its price is a significant factor. Understanding this relationship helps to know what people value in a home and how much they're willing to pay for it.

[ ]:
```
fig, axs = plt.subplots(1, 4, figsize=(20, 5))

# Scatter plot for 'house_size'
axs[0].scatter(df['house_size'], df['price'], color="purple", alpha=0.5)
axs[0].set_xlabel("House Size")
axs[0].set_ylabel("House Price")
axs[0].set_title("Price vs House Size")

# Scatter plot for 'bed'
axs[1].scatter(df['bed'], df['price'], color="purple", alpha=0.5)
axs[1].set_xlabel("Bed")
axs[1].set_ylabel("House Price")
axs[1].set_title("Price vs Bed")

# Scatter plot for 'bath'
axs[2].scatter(df['bath'], df['price'], color="purple", alpha=0.5)
axs[2].set_xlabel("Bed")
axs[2].set_ylabel("House Price")
axs[2].set_title("Price vs Bath")

# Scatter plot for 'acre_lot'
axs[3].scatter(df['house_size'], df['price'], color="purple", alpha=0.5)
axs[3].set_xlabel("Acre Lot")
axs[3].set_ylabel("House Price")
axs[3].set_title("Price vs House Size")

fig.tight_layout()
plt.show()
```



The scatter plots reveal a stronger positive correlation between the number of bedrooms and bathrooms with housing price compared to the influence of house sizes. Tighter clustering and an upward trend indicate that higher bedroom and bathroom counrs for the highe values of properties, emphasizing their significant impact on real estate prices.

[ ]:
```
column_num = ['bed','bath','acre_lot','house_size','price']
Q1 = df[column_num].quantile(0.25)
Q3 = df[column_num].quantile(0.75)
IQR = Q3 - Q1

df = df[~((df[column_num] < (Q1 - 1.5 * IQR)) | (df[column_num] > (Q3 + 1.5 *␣
↪IQR))).any(axis=1)]
```

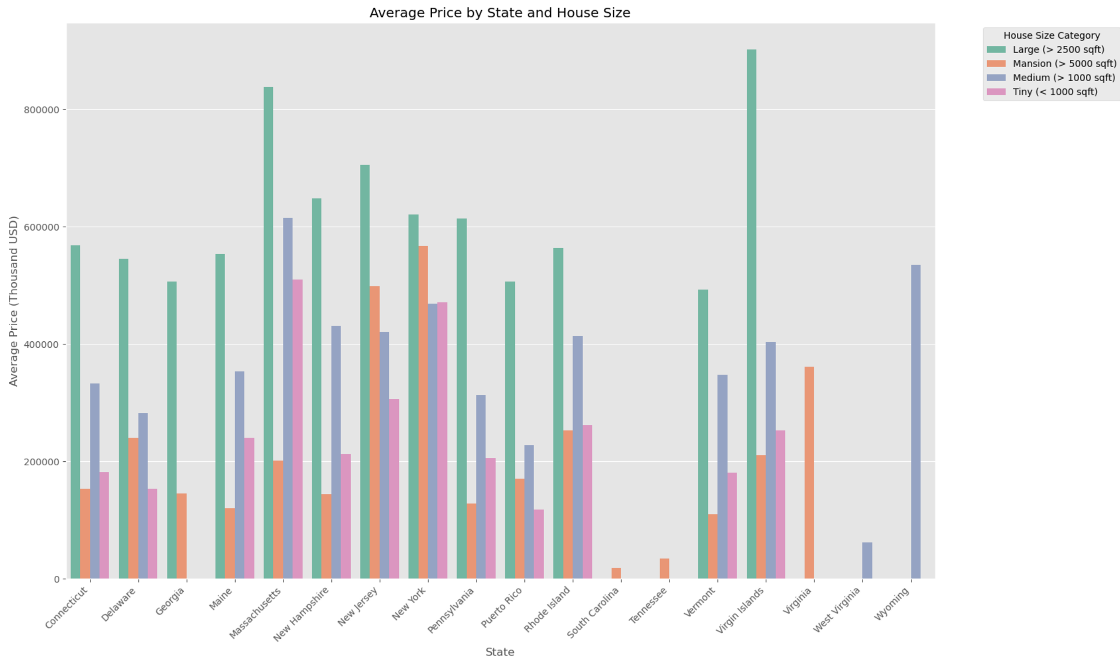
[ ]:
```
df_mean = df.groupby('acre_lot')['price'].mean().reset_index()
fig = px.scatter(df_mean, x='acre_lot', y='price', trendline='ols')
fig.update_layout(title='Average Price per Acre Lot', xaxis_title='Acre Lot',␣
↪yaxis_title='Price')
fig.show()
```

[ ]:
```
city_bed_count = df.groupby(['state', 'bath']).size().reset_index(name='count')
city_bed_count = city_bed_count.sort_values(by='count', ascending=False)
fig = px.bar(city_bed_count, x='state', y='count', color='bath',␣
↪barmode='stack',
             title='Distribution of State with Bath',color_discrete_sequence=px.
↪colors.qualitative.Set2)
fig.show()
```

The distribution of bedroom and bathroom counts illustrates a clear concentration of properties with higher number in large states like New York and Massachusetts. This aligns with the presence of multiple metropolitan areas and cities, attracting a skilled labor force with increased housing demand, potentially contributing to elevated housing prices in these states.

[ ]:
```
# Create a square_feet function to categorize houses based on their sizes
def square_feet(house_size):
    if house_size <= 1000:
        return 'Tiny (< 1000 sqft)'
    elif 1000 < house_size <= 2500:
```

```
        return 'Medium (> 1000 sqft)'
    elif 2500 < house_size <= 5000:
        return 'Large (> 2500 sqft)'
    else:
        return 'Mansion (> 5000 sqft)'

# Apply the square_feet function to create a new column 'house_size_category'
df['house_size_category'] = df['house_size'].apply(square_feet)

# Calculate the average price for each category of houses and group the data
grouped_house_sizes = df.groupby(['state', 'house_size_category'])['price'].
 ↪mean().reset_index()

plt.figure(figsize=(20, 10))

# Plot the grouped bar chart with additional separation
sns.barplot(x='state', y='price', hue='house_size_category',␣
 ↪data=grouped_house_sizes, palette="Set2", dodge=1)

plt.title('Average Price by State and House Size')
plt.xlabel('State')
plt.ylabel('Average Price (Thousand USD)')

plt.xticks(rotation=45, ha='right')
plt.legend(title='House Size Category', bbox_to_anchor=(1.05, 1), loc='upper␣
 ↪left')

plt.tight_layout(rect=[0, 0, 0.85, 1])
plt.show()
```

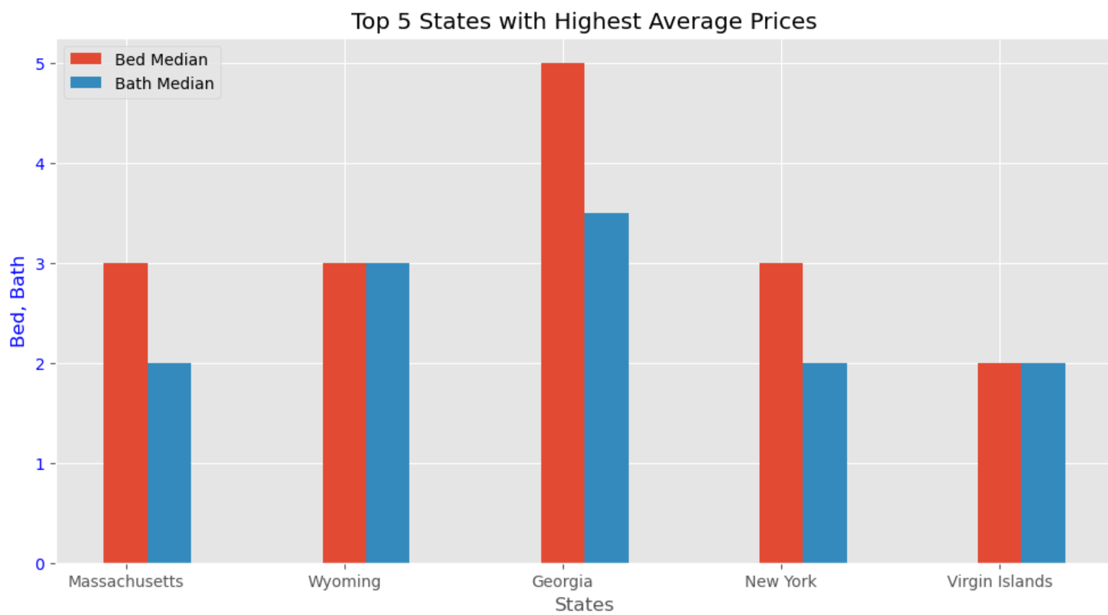


From the chart, New York and Massachusetts stand out for recording high prices in the real estate market, particularly for large houses with sizes exceeding 2500 square feet. These states are home to major cities that serve as centers of economic activity and education. Virgin Island, on the other hand, has a severely limited supply of housing with very high cost of living due to imported goods.

[ ]:
```
import matplotlib.pyplot as plt
from mpl_axes_aligner import align

top5=['Massachusetts', 'Wyoming', 'Georgia', 'New York', 'Virgin Islands']
filtered_data = df[df['state'].isin(top5)]

data_by_state = [filtered_data[filtered_data['state'] == state]['house_size'].
 ↪values for state in top5]

# Select the top 5 states with the highest average prices
top_5_states = grouped_by_state.head(5)

# Extract relevant data for plotting
states = top_5_states['state_name']
bed_medians = top_5_states['bed_median']
bath_medians = top_5_states['bath_median']
house_size_medians = top_5_states['house_size_median']

# Set up the figure and axes
fig, ax1 = plt.subplots(figsize=(12, 6))
```

```
# Plotting the bar chart for Bed Median and Bath Median
bar_width = 0.2
index = range(len(states))
bar2 = ax1.bar([i + bar_width for i in index], bed_medians, width=bar_width,␣
 ↪label='Bed Median')
bar3 = ax1.bar([i + 2 * bar_width for i in index], bath_medians,␣
 ↪width=bar_width, label='Bath Median')

# Set labels and title for the first y-axis
ax1.set_xlabel('States')
ax1.set_ylabel('Bed, Bath', color='blue')
ax1.set_title('Top 5 States with Highest Average Prices')
ax1.set_xticks([i + bar_width for i in index])
ax1.set_xticklabels(states)
ax1.tick_params(axis='y', labelcolor='blue')
ax1.legend(loc='upper left')
```

```
[ ]: <matplotlib.legend.Legend at 0x282cdba10>
```



1.4 Mapping

To visualize the average housing prices across states on a map, it is essential to define the longitude and latitude coordinates. The geographic data utilized for this project is sourced from the US Census Bureau.

First, the average and standard deviation of housing prices and arce_lot are calulated for each state.

[ ]:
```python
# Grouping the data by state and calculating the mean and standard deviation of
↪price and acre_lot
grouped_by_state = df.groupby(['state']).agg(
    price_mean=('price', 'mean'),
    price_median=('price', 'median'),
    acre_lot_mean=('acre_lot', 'mean'),
    acre_lot_std=('price', 'std'),
    bed_median=('bed', 'median'),
    bath_median=('bath', 'median'),
    house_size_median=('house_size', 'median')
)
grouped_by_state['state_name'] = grouped_by_state.index
grouped_by_state = grouped_by_state.round(2)
# Presenting the result in descending order
grouped_by_state = grouped_by_state.sort_values(by='price_mean',
↪ascending=False)
grouped_by_state['state_name'] = grouped_by_state.index
grouped_by_state['state_abbrev'] = grouped_by_state['state_name'].
↪map(abbreviation_mapping)
grouped_by_state
```

[ ]:
```
                 price_mean  price_median  acre_lot_mean  acre_lot_std  \
state
Massachusetts     584805.99      545000.0           0.55     345638.87
Wyoming           535000.00      535000.0           0.29          0.00
New York          513221.15      435000.0           0.47     367532.99
Georgia           492703.60      490225.0           0.91      74803.07
New Jersey        474499.09      425000.0           0.33     269153.78
Rhode Island      412676.16      350000.0           0.39     222846.38
New Hampshire     371170.51      330000.0           1.02     269835.07
Virginia          362064.52      249000.0           0.28     272236.04
Connecticut       340303.74      279900.0           0.70     231905.84
Pennsylvania      317892.98      269900.0           0.25     231170.66
Delaware          314051.61      275000.0           0.19     186445.75
Maine             301959.04      225000.0           1.00     267824.30
Vermont           287959.76      225000.0           1.04     250852.66
Virgin Islands    248588.73      165000.0           0.83     266080.74
Puerto Rico       220407.13      128000.0           0.30     250310.10
West Virginia      62500.00       62500.0           0.17          0.00
Tennessee          34900.00       34900.0           0.92          0.00
South Carolina     18950.00       18950.0            NaN          0.00

                 bed_median  bath_median  house_size_median      state_name  \
state
Massachusetts           3.0          2.0             1550.0   Massachusetts
Wyoming                 3.0          3.0             1935.0         Wyoming
New York                3.0          2.0             1488.0        New York
```

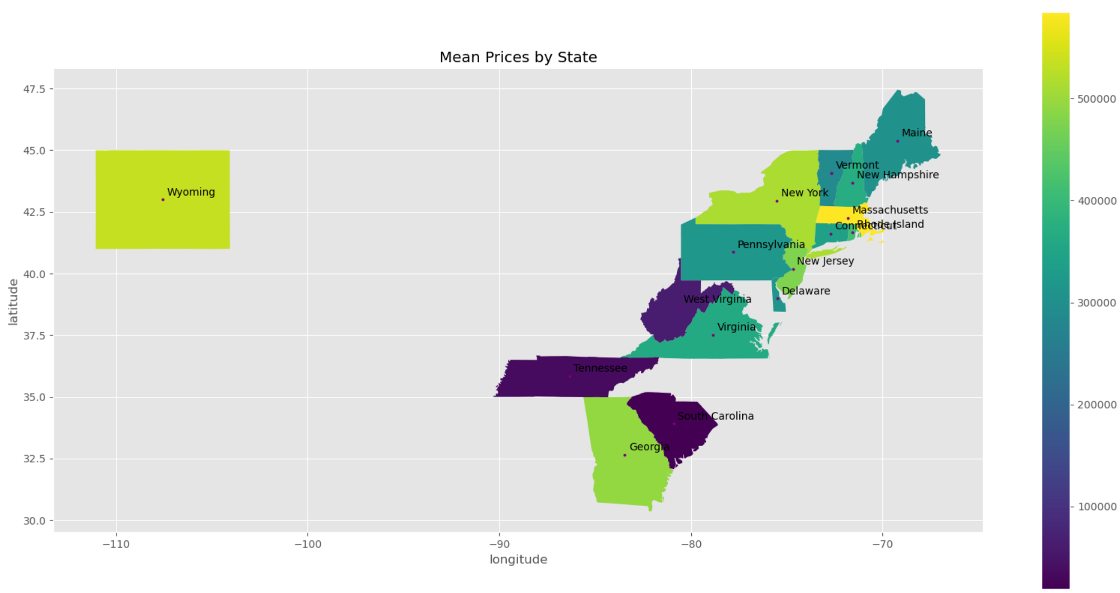
```
Georgia             5.0        3.5        3388.5         Georgia
New Jersey          3.0        2.0        1551.0      New Jersey
Rhode Island        3.0        2.0        1488.0    Rhode Island
New Hampshire       3.0        2.0        1765.0   New Hampshire
Virginia            NaN        NaN           NaN        Virginia
Connecticut         3.0        2.0        1574.0     Connecticut
Pennsylvania        3.0        2.0        1440.5    Pennsylvania
Delaware            3.0        2.0        1750.0        Delaware
Maine               3.0        2.0        1562.0           Maine
Vermont             3.0        2.0        1700.0         Vermont
Virgin Islands      3.0        2.0        1326.0  Virgin Islands
Puerto Rico         3.0        2.0        1250.0     Puerto Rico
West Virginia       4.0        2.0        1860.0   West Virginia
Tennessee           NaN        NaN           NaN       Tennessee
South Carolina      NaN        NaN           NaN  South Carolina

               state_abbrev
state
Massachusetts            MA
Wyoming                  WY
New York                 NY
Georgia                  GA
New Jersey               NJ
Rhode Island             RI
New Hampshire            NH
Virginia                 VA
Connecticut              CT
Pennsylvania             PA
Delaware                 DE
Maine                    ME
Vermont                  VT
Virgin Islands          NaN
Puerto Rico              PR
West Virginia            WV
Tennessee                TN
South Carolina           SC
```

Now, the map is generated.

[ ]:
```python
import geopandas as gpd

fig, gax = plt.subplots(figsize=(20,10))
# Create a second DataFrame with geometric data from US Census Bureau
df_states = gpd.read_file('/Users/hanhhieudao/Downloads/states_21basic/states.
↪shp')

# Merge the two DataFrames based on the 'state_name' column
```

```
merged_df = pd.merge(df_states, grouped_by_state, left_on='STATE_NAME', right_on='state_name', how='right')
merged_df = merged_df.dropna(subset=['price_mean'])

# Plot the merged DataFrame
merged_df.plot(ax=gax, column='price_mean', legend=True, cmap='viridis')
merged_df['geometry'] = merged_df['geometry'].centroid

# Plot the state centroids as purple dots
merged_df.plot(ax=gax, color='purple', markersize=5)

gax.set_xlabel('longitude')
gax.set_ylabel('latitude')
plt.title('Mean Prices by State')

for x, y, label in zip(merged_df['geometry'].x, merged_df['geometry'].y, merged_df['state_name']):
    gax.annotate(label, xy=(x,y), xytext=(4,4), textcoords='offset points')

gax.spines['top'].set_visible(False)
gax.spines['right'].set_visible(False)
plt.show()
```

```
/var/folders/sc/hghgdjk51pbdrq1vbp6569q00000gn/T/ipykernel_17477/2619246026.py:1
3: UserWarning: Geometry is in a geographic CRS. Results from 'centroid' are
likely incorrect. Use 'GeoSeries.to_crs()' to re-project geometries to a
projected CRS before this operation.
```



The map visualization distinctly portrays Massachusetts and New York with notably high housing prices, indicating strong real estate markets likely driven by urban demand, economic activity, and social demographics. In contrast, West Virginia emerges as a region with comparatively lower housing prices, potentially reflecting a distinct economic landscape and lower demand dynamics within the state's housing market.

```
[ ]: # Add longitue and lattitude for each state
from geopy.geocoders import Nominatim
from shapely.geometry import Point

# Create a geolocator object
geolocator = Nominatim(user_agent="my_geocoder")

def get_lat_long(location):
    try:
        location = geolocator.geocode(location)
        return location.latitude, location.longitude
    except:
        return None, None

# Apply the function to each state in the dataset
grouped_by_state['latitude'], grouped_by_state['longitude'] =␣
 ↪zip(*grouped_by_state['state_name'].apply(get_lat_long))
grouped_by_state['Coordinates'] = list(zip(grouped_by_state.longitude,␣
 ↪grouped_by_state.latitude))
grouped_by_state["Coordinates"] = grouped_by_state["Coordinates"].apply(Point)
```

1.5 Merging US Real Estate Datase with population Demographic Dataset

```
[ ]: file_path = '/Users/hanhhieudao/Desktop/ECO225/DECENNIALDP2020.DP1-Data.csv'
df_population = pd.read_csv(file_path, skiprows=[1])
df_population.head()
```

```
[ ]:        GEO_ID        NAME  DP1_0001C  DP1_0002C  DP1_0003C  DP1_0004C  \
0  0400000US01     Alabama    5024279     286529     302637     325031
1  0400000US02      Alaska     733391      48104      51054      51344
2  0400000US04     Arizona    7151502     392370     443878     485297
3  0400000US05    Arkansas    3011524     179575     192794     205837
4  0400000US06  California   39538223    2137439    2393219    2613891

   DP1_0005C  DP1_0006C  DP1_0007C  DP1_0008C  …  DP1_0152P  DP1_0153P  \
0     338475     345931     314244     311116  …        1.0        0.5
1      47433      49456      55058      56981  …        1.0        0.4
2     485891     477713     473578     462909  …        1.0        0.4
3     204915     198109     188836     190366  …        1.1        0.7
4    2644071    2731553    2915258    2911574  …        0.5        0.2
```

```
   DP1_0154P  DP1_0155P  DP1_0156P  DP1_0157P  DP1_0158P  DP1_0159P  \
0        2.5        4.6        (X)        (X)      100.0       67.7
1        9.1        3.6        (X)        (X)      100.0       63.9
2        5.9        1.7        (X)        (X)      100.0       65.3
3        2.4        4.3        (X)        (X)      100.0       65.0
4        2.1        1.3        (X)        (X)      100.0       54.5

   DP1_0160P  Unnamed: 322
0       32.3           NaN
1       36.1           NaN
2       34.7           NaN
3       35.0           NaN
4       45.5           NaN

[5 rows x 323 columns]
```

[ ]:
```python
population = df_population[['NAME', 'DP1_0001C', 'DP1_0025C', 'DP1_0049C',␣
 ↪'DP1_0086C', 'DP1_0087C', 'DP1_0089C', 'DP1_0090C']]
merged = pd.merge(population, grouped_by_state, left_on='NAME',␣
 ↪right_on='state_name', how='inner')
merged = merged.rename(columns={'DP1_0001C': 'total_population'})
merged= merged.rename(columns={'DP1_0025C': 'male'})
merged = merged.rename(columns={'DP1_0049C': 'female'})
merged = merged.rename(columns={'DP1_0086C': 'White'})
merged = merged.rename(columns={'DP1_0087C': 'Black/African'})
merged = merged.rename(columns={'DP1_0089C': 'Asian'})
merged = merged.rename(columns={'DP1_0090C': 'Hawaiian'})
merged.dropna()
merged.head()
```

[ ]:
```
            NAME  total_population     male   female    White  Black/African  \
0    Connecticut           3605944  1749853  1856091  2692022         467416
1       Delaware            989948   476719   513229   665198         244944
2        Georgia          10711908  5188570  5523338  6212741        3538146
3          Maine           1362359   667560   694799  1299963          36304
4  Massachusetts           7029917  3401702  3628215  5399122         669866

    Asian  Hawaiian  price_mean  price_median  acre_lot_mean  acre_lot_std  \
0  205693      5971   340303.74      279900.0           0.70     231905.84
1   50969      1547   314051.61      275000.0           0.19     186445.75
2  565644     19020   492703.60      490225.0           0.91      74803.07
3   25473      1619   301959.04      225000.0           1.00     267824.30
4  582484     10436   584805.99      545000.0           0.55     345638.87

   bed_median  bath_median  house_size_median   state_name state_abbrev  \
0         3.0          2.0             1574.0  Connecticut           CT
1         3.0          2.0             1750.0     Delaware           DE
```

```
2          5.0          3.5             3388.5        Georgia            GA
3          3.0          2.0             1562.0          Maine            ME
4          3.0          2.0             1550.0  Massachusetts            MA

    latitude  longitude                      Coordinates
0  41.650020 -72.734216  POINT (-72.7342163 41.6500201)
1  38.692045 -75.401331  POINT (-75.4013315 38.6920451)
2  32.329381 -83.113737  POINT (-83.1137366 32.3293809)
3  45.709097 -68.859020   POINT (-68.8590201 45.709097)
4  42.378877 -72.032366   POINT (-72.032366 42.3788774)
```

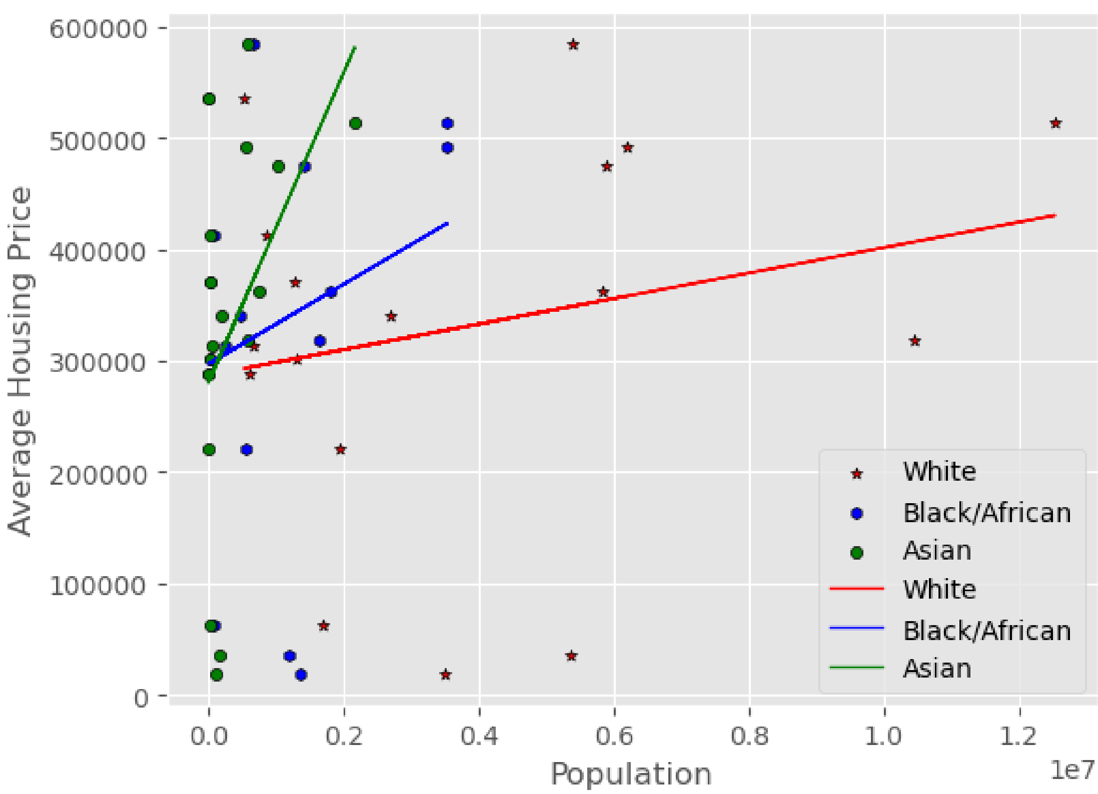
[ ]:
```
fig = px.scatter_mapbox(merged,
                        lat='latitude',
                        lon='longitude',
                        size='total_population',
                        size_max=20,  # Set the maximum size of bubbles for␣
 ↪'male'
                        color='price_mean',
                        zoom=6,
                        center=dict(lat=42.3601, lon=-71.0589),
                        text='state_name',
                        mapbox_style='carto-positron'
                       )

fig.show()
```

1.6 The Message

The plot compares the average housing prices for different racial groups based on their population sizes in 13 states collected from the dataset of US Real Estate Market. It reveals the trends in housing prices concerning population sizes, with regression lines highlighting the correlation with White, Black/African, and Asian groups.

[ ]:
```
# Create a figure and axis
fig, ax = plt.subplots()
x3 = merged['White']
y3 = merged['price_mean']
x4 = merged['Black/African']
y4 = merged['price_mean']
x5 = merged['Hawaiian']
y5 = merged['price_mean']
x6 = merged['Asian']
y6 = merged['price_mean']

# Scatter plots
ax.scatter(x3, y3, c="red", marker="*", edgecolors='black', s=20, label='White')
```

```
ax.scatter(x4, y4, c="blue", marker="h", edgecolors='black', s=20, label='Black/
↪African')
ax.scatter(x6, y6, c="green", marker="o", edgecolors='black', s=20,␣
↪label='Asian')

#Regression lines
m3, b3 = np.polyfit(x3, y3, 1)
plt.plot(x3, m3*x3+b3, color='red', linewidth=1, label='White')
m4, b4 = np.polyfit(x4, y4, 1)
plt.plot(x4, m4*x4+b4, color='blue', linewidth=1, label='Black/African')
m6, b6 = np.polyfit(x6, y6, 1)
plt.plot(x6, m6*x6+b6, color='green', linewidth=1, label='Asian')

ax.set_xlabel("Population")
ax.set_ylabel("Average Housing Price")

ax.legend()
plt.show()
```



From the plotted regression lines, the Asian community exhibits the steepest slope, indicating a highest correlation of population size with housing prices. This finding suggests further exploration into the potential social factors contributing to the impact of Asian group on the US Real Estate

Market. Some possible determinants include household income and employment rate can bright meaningful insights of their purchasing power and its effect on US property demand and prices.

Following the Asian population, the Black/African group displays a moderately steep slope, indicative of a positive correlation with housing prices. Conversely, the regression line for the White population demonstrates a gradual slope, suggesting a comparatively weaker correlation between White population size and housing prices.

Notably, despite the lower population size of the Asian community in most states compared to the White population, the housing prices associated with Asian group are marked higher. This observation indicates a deeper study on the disproportionate impact of Asian demographic on the US real estate market.

2 Conclusion

In this paper, I analyze the contribution of key determinants of properties to investigate how they affect their prices in the US Real Estate market. With a large database of properties across multiple states in the US, this paper ressearched whether there is a specific trends or patterns in the prices regarding to the property's characteristics based on collected data.

The rising housing prices are linked to the ammentites of the properties, including their sizes, number of rooms, and location. In addition, the fluctuation in housing prices also reflect the diversity of demographic buyers with distinct backgrounds. Moreover, a comprehensive analysis of US real estate markey not only represents the trends in housing prices, but also reflects broafer dynamics of the US economy, encapsulating the diverse demands, preferences, and income levels of various social groups.

2.1 References

- Bayer, P., Casey, M., Ferreira, F., & McMillan, R. (Year). Racial and ethnic price differentials in the housing market. Journal Name, Volume(Issue), PageRange.
- U.S. Census Bureau. (2020). 2020 Census Demographic Profile. U.S. Census Bureau.https://www.census.gov/data/tables/2023/dec/2020-census-demographic-profile.html
- Realtor.com. (2024). Real Estate Listings in the US by State and Zip Code. Kaggle.