Predicting California Housing Prices

Mikhail Zaatra
Dawit Daniel Alaro
Srividhya Thirumalairajan
Trong Quyen Nguyen

AGENDA

- Project Overview:
  - Topic and motivation
  - Project goals: Questions the team hopes to answer.
- Technologies, languages, tools, and algorithms used throughout the project
- Description of the source of data
- Data exploration & transformation
- Project analysis
- Final Project outcome , Dashboard and visualization
- Recommendation for future analysis

PROJECT OVERVIEW

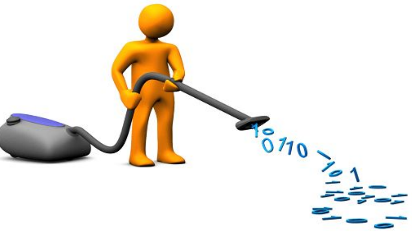
- Motivation:
  - Housing prices are a hot topic, especially during the COVID-19 Pandemic
  - Our group is passionate about real estate investments and wanted to build Machine Learning Model to help homeowners and real estate investors to evaluate potential deals within California
- Project overview:
  - We imported housing sale records as CSV formats, then we used Python Libraries and SQL to perform ETL process on the raw data. Once data has been loaded we built a supervised Linear regression and Neural network MLs to predict the housing prices.
- Project goals: Questions the team hopes to answer:
  - Can we predict the average housing prices and help consumers and real estate investors to make educated decisions based on our results and predicted housing prices?

TECHNOLOGIES, LANGUAGES, TOOLS, AND ALGORITHMS USED THROUGHOUT THE PROJECT

- Python : Pandas, PySpark
- SQLAlchemy
- Postgres
- Google CoLab
- Google Docs.
- Python ML Models: Linear regression , Keras for neural network ML Model.

Data Exploration and Transformation

Upload the Raw Data via AWS	Data Wrangling	Results	Visualization & Analysis

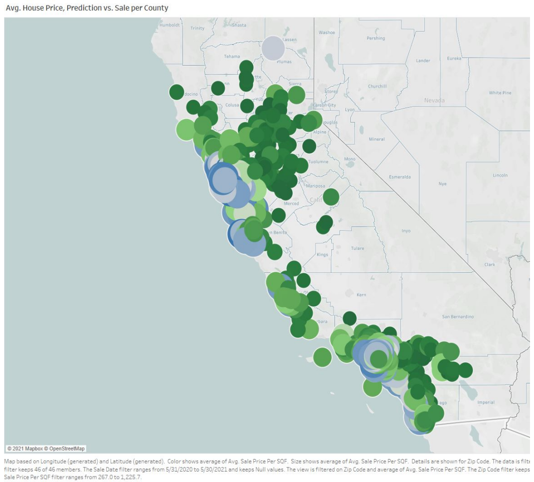


Data Wrangling

- Cleaned and unified the messy and complex data sets for easy access and analysis

Results

- Able to produce following clean data sets for our machine learning models
- Final_data.csv, house_data.csv, Sale_date.csv

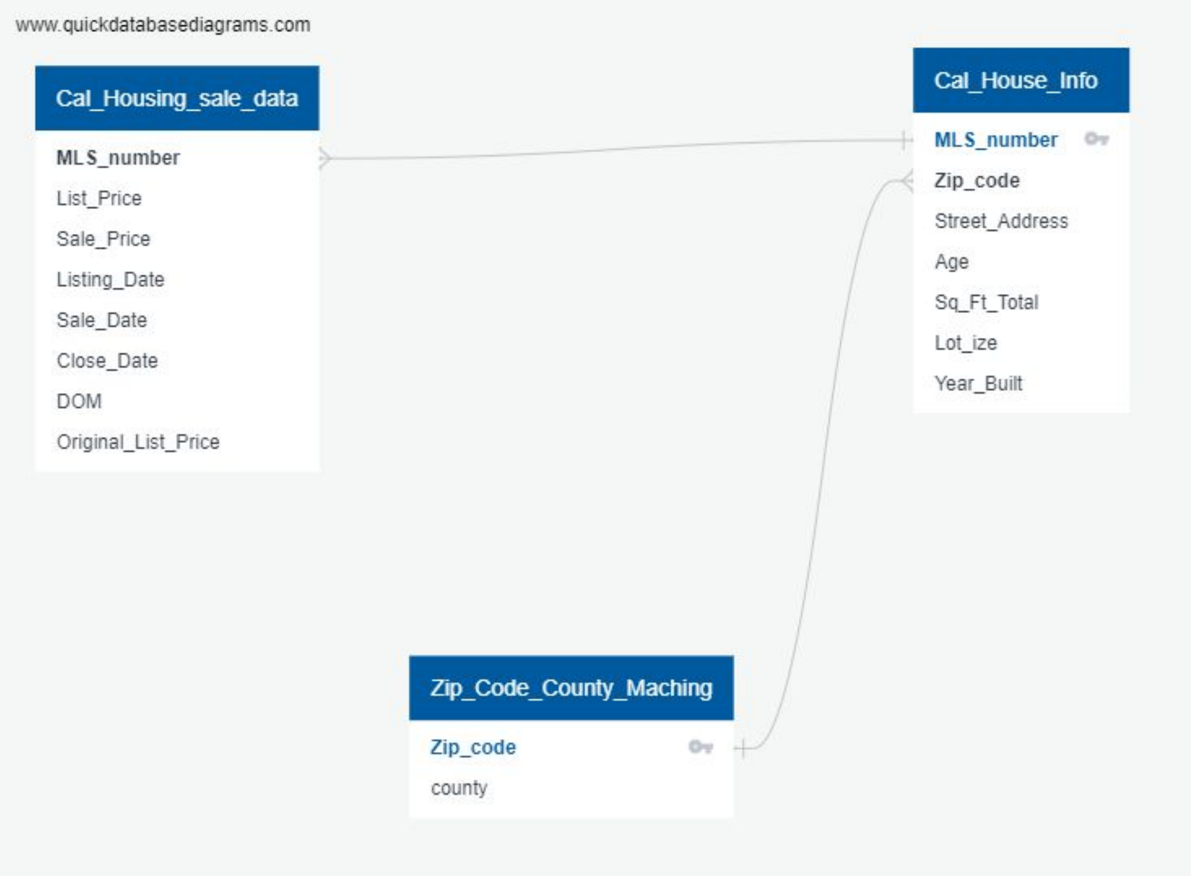


Data Processing Method

- Build ERDs

- create new tables
  - Houseing_Sale_data
  - House_info
  - County_info

Dawit Alaro



Data Processing

- Pyspark and AWS
- Transform String columns
- Merge
- Select columns
- Drop nulls values
- Lot size >= 800 SqFt
- Load to PostgreSql(three table)

Dawit Alaro

Clean Data with Regression prediction

	County_Index	SqFtTotal	Lot_Size	Age	BedsTotal	BathsTotal	DOM	Year_Sold	List_Price	Sale_Price	Prediction
0	4.0	1504.0	10019.0	58.0	3.0	2.0	4.0	2021.0	459000.0	502000.0	4.874488e+05
1	14.0	1862.0	5850.0	60.0	4.0	3.0	9.0	2021.0	725000.0	740000.0	7.621011e+05
2	0.0	1917.0	5341.0	56.0	3.0	3.0	8.0	2021.0	1349000.0	1500000.0	1.405694e+06
3	0.0	3857.0	11019.0	33.0	5.0	4.0	9.0	2021.0	1495000.0	1608000.0	1.560025e+06
4	17.0	1840.0	8008.0	63.0	3.0	2.0	28.0	2021.0	939900.0	945000.0	9.840122e+05
...	...	...	...	...	...	...	...	...	...	...	...
81	2.0	1964.0	6991.0	57.0	4.0	2.0	9.0	2021.0	1085000.0	1078500.0	1.133644e+06
82	2.0	1231.0	6114.0	66.0	4.0	2.0	14.0	2021.0	650000.0	720000.0	6.836707e+05
83	0.0	1136.0	12427.0	93.0	3.0	1.0	0.0	2021.0	545000.0	550000.0	5.759723e+05
84	6.0	2115.0	11773.0	37.0	4.0	3.0	5.0	2021.0	875000.0	905000.0	9.176531e+05
85	15.0	3935.0	21449.0	112.0	5.0	4.0	13.0	2021.0	7595000.0	7600000.0	7.854004e+06

86 rows × 11 columns

Connection, Regression, Neural network, Random Forest

TrongQuyen Nguyen

```
# Need to make some decision here, which will effect all the 4 models
test_size = 0.05 # due to sample size of 4K thousand sample, choose something like 0.05;
random_state = 6
hidden_nodes_layer1 = 15 # Change this number will affect both NN and NN2 models
hidden_nodes_layer2 = 20 # Change this number will affect both NN and NN2 models
hidden_nodes_layer3 = 10 # Change this number will affect both NN and NN2 models
activation='relu'
activation_last='linear'
loss_input='mean_absolute_error'
optimizer_input='Adam'
metrics_input='MSE'
size_batch_no = 32
epochs_no = 200
```

```
# Split training/test datasets
# Regression 1 and Neural Network 1 need X_train, not X2
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size

# Model 2 exclude List_Price amongst the independent variables
# Regression 2 and Neural Network 2 "EXCLUDEs" the variable "List_
X2_train = X_train.drop(columns=['List_Price']) # This way, we can
X2_test = X_test.drop(columns=['List_Price']) # The same, the numbe
```

Connection, Regression, Neural network, Random Forest

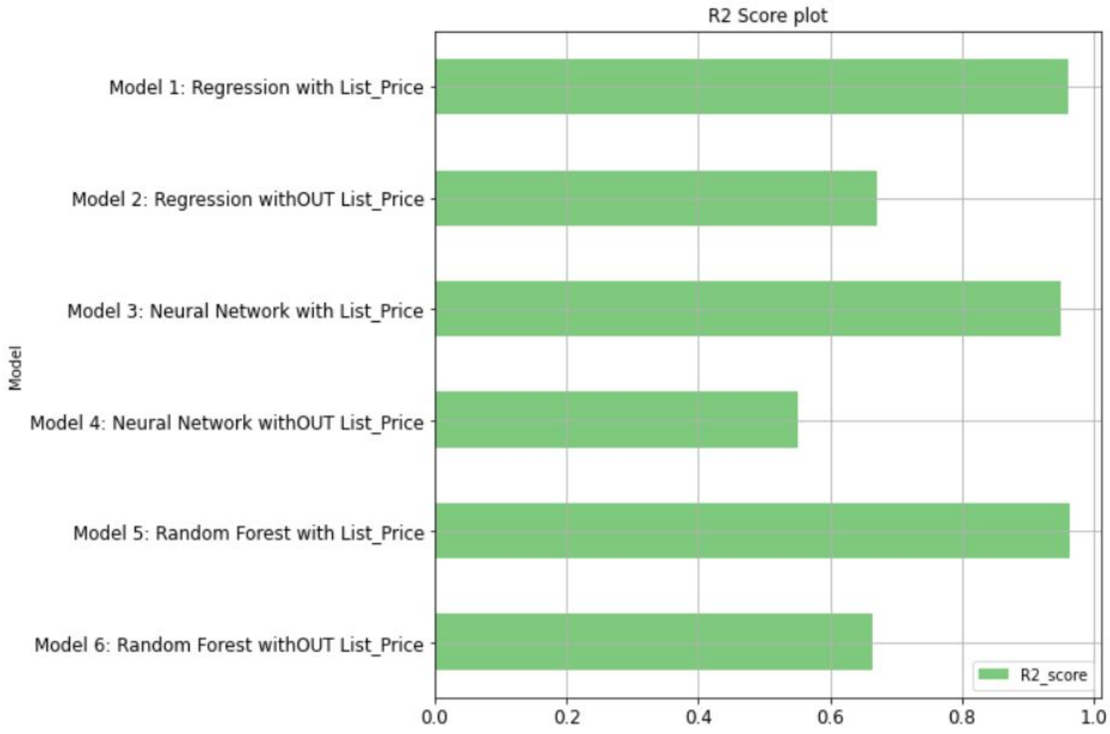
TrongQuyen Nguyen

Examples of the prediction values

	Sale_Price	Predict_Reg_1	Predict_Reg_2	Predict_NN_3	Predict_NN_4	Predict_rfr_5	Predict_rfr_6
0	1005000	913,281.32	1,416,121.20	[935181.6]	[821084.75]	936,562.73	1,573,790.00
1	1175000	1,219,022.44	1,559,513.55	[1233525.8]	[1347218.4]	1,189,697.00	1,318,052.00
2	565000	587,771.36	639,187.61	[597232.06]	[673489.5]	569,100.00	508,025.00
3	950000	959,388.14	1,152,340.72	[973758.9]	[994087.1]	946,985.91	1,076,372.50
4	625000	608,544.68	526,425.80	[592482.94]	[634491.8]	596,205.64	723,884.00
5	1925000	1,872,278.90	965,161.08	[1881650.1]	[981521.25]	1,952,923.48	845,301.19
6	21150000	25,521,630.17	10,581,912.84	[26075766.0]	[10951954.0]	17,499,100.00	11,077,950.00
7	2000000	1,830,298.75	1,322,725.73	[1776446.0]	[1154661.8]	1,936,584.78	1,473,238.85
8	341000	321,158.59	564,749.91	[318526.56]	[731683.0]	306,590.00	573,271.00
9	1450000	1,469,391.98	2,446,670.51	[1520088.6]	[1722921.4]	1,542,606.05	1,543,560.00

Connection, Regression, Neural network, Random Forest

TrongQuyen Nguyen



With “List_Price”, R2_score is the higher than without

Connection, Regression, Neural network, Random Forest

TrongQuyen Nguyen

	count	mean	std	min	25%	50%	75%	max
SqFtTotal	4225.0	2.005063e+03	9.413365e+02	454.0	1368.0	1792.0	2383.0	13300.0
Lot_Size	4225.0	3.089252e+05	9.476886e+06	864.0	5683.0	7149.0	10063.0	493360560.0
Age	4225.0	4.465278e+01	2.580530e+01	0.0	23.0	43.0	64.0	138.0
BathsTotal	4225.0	2.480710e+00	1.010442e+00	0.0	2.0	2.0	3.0	12.0
BedsTotal	4225.0	3.438817e+00	8.884830e-01	1.0	3.0	3.0	4.0	8.0
BathsFull	4225.0	2.208757e+00	8.359631e-01	0.0	2.0	2.0	3.0	8.0
BathsHalf	4225.0	2.719527e-01	4.627524e-01	0.0	0.0	0.0	1.0	4.0
DOM	4225.0	1.036047e+01	3.052176e+01	0.0	3.0	6.0	9.0	1013.0
Year_Sold	4225.0	2.020981e+03	1.371441e-01	2020.0	2021.0	2021.0	2021.0	2021.0
List_Price	4225.0	1.014835e+06	1.119410e+06	76900.0	499000.0	715900.0	1130000.0	24999000.0

‘List_Price’ as a quality parameter

KEY TAKEAWAYS AND RECOMMENDATIONS

- Summary:
    - In this project we created a powerful tool for Reals estate investors and potential home Buyers to Make educated and Data Driven Decisions when it comes to real estate Purchase:
        - They Can decide in which county & City they want to live (Based on Affordability).
        - Then they can decide on the Size of the House & Number of Bedrooms.
    - Using Linear Regression to predict Housing Prices provides good prediction with over 90% R - squared value.
- Recommendations:
    - For Future projects, We could enhance the housing price prediction Model by using a deep learning model that uses the the time as Independent variable
    - We could extended this Model to Take Household Income, school Rating, and other layer that might impact the Housing Prices.
    - This model could be extended to be in use Nationwide. We could up level it to start from the state then drill down to county and they to the City level. .

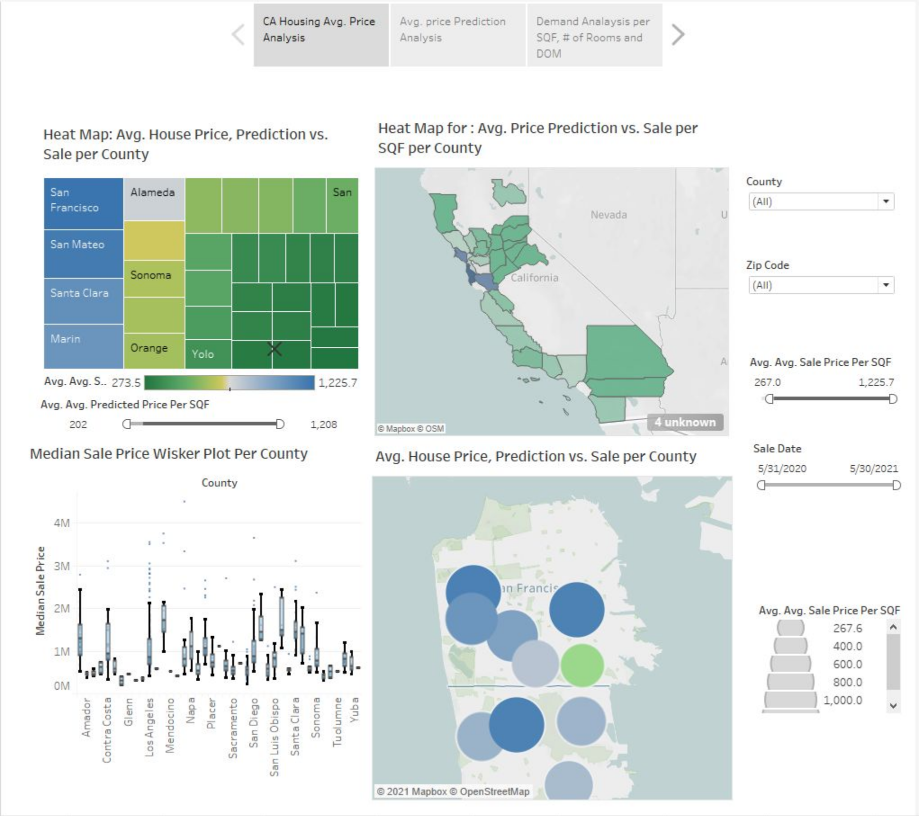
Thank you!

DASHBOARD AND VISUALIZATION USING TABLEAU

HOUSING HEAT MAPS PER AvG. PRICES

The following Dashboard present:

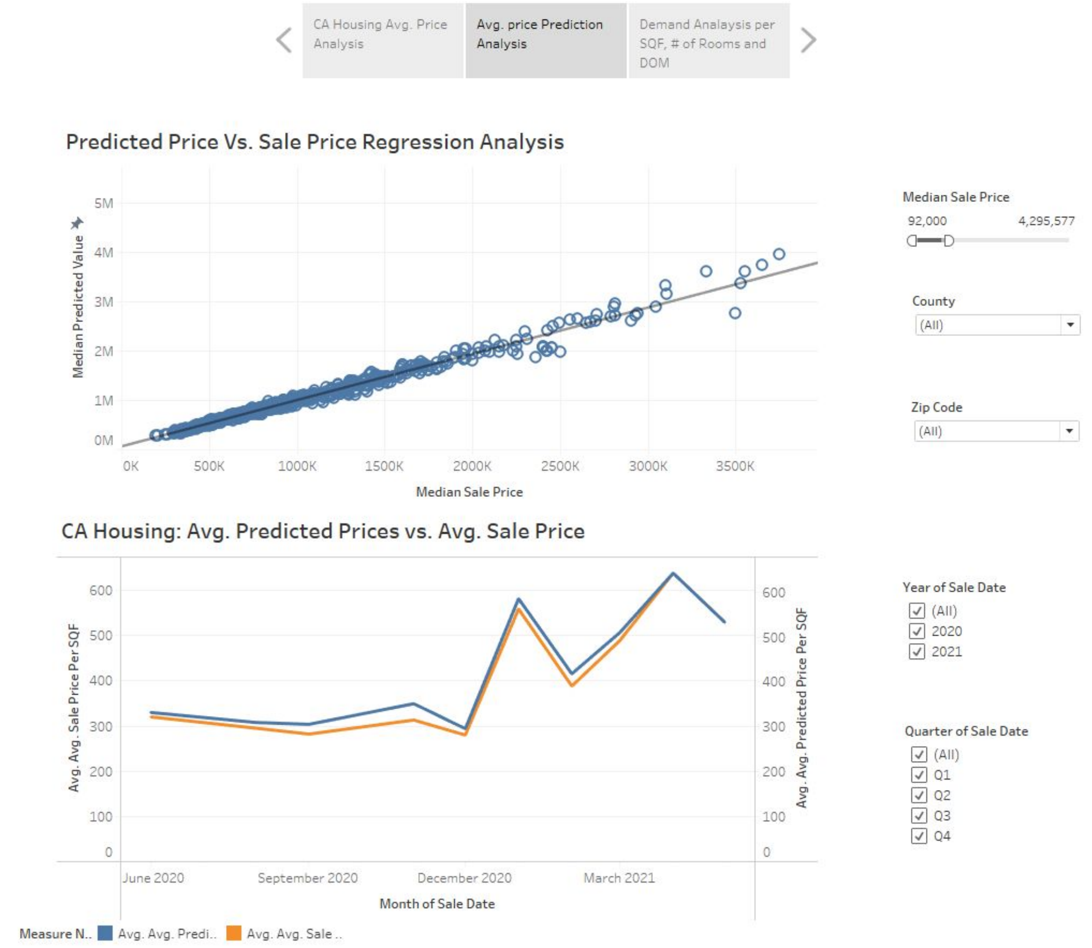
1. Heat Map for Avg. price Per County (Blue is the Highest Avg. Price).
2. Heat Map of Avg. Price Per SQF for each County
3. Heat Map of Avg. Price Per SQF for each Zip Code
4. Whisker Plot chart to show the Price distribution between the 4 quartiles



HOUSE PRICING REGRESSION CHARTS

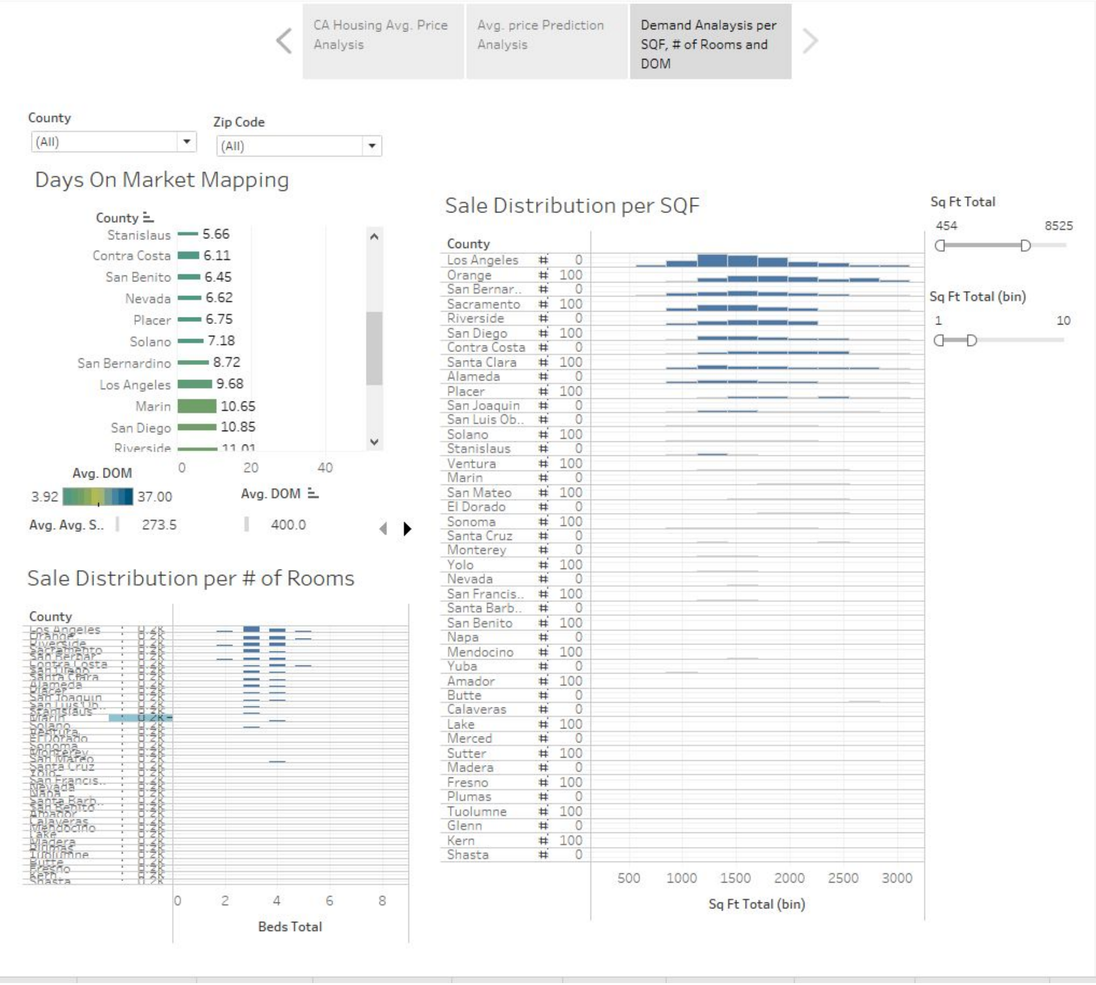
In this Dashboard present :

1) Sale Price vs. Predicted Price Based on the Linear Regression Model.



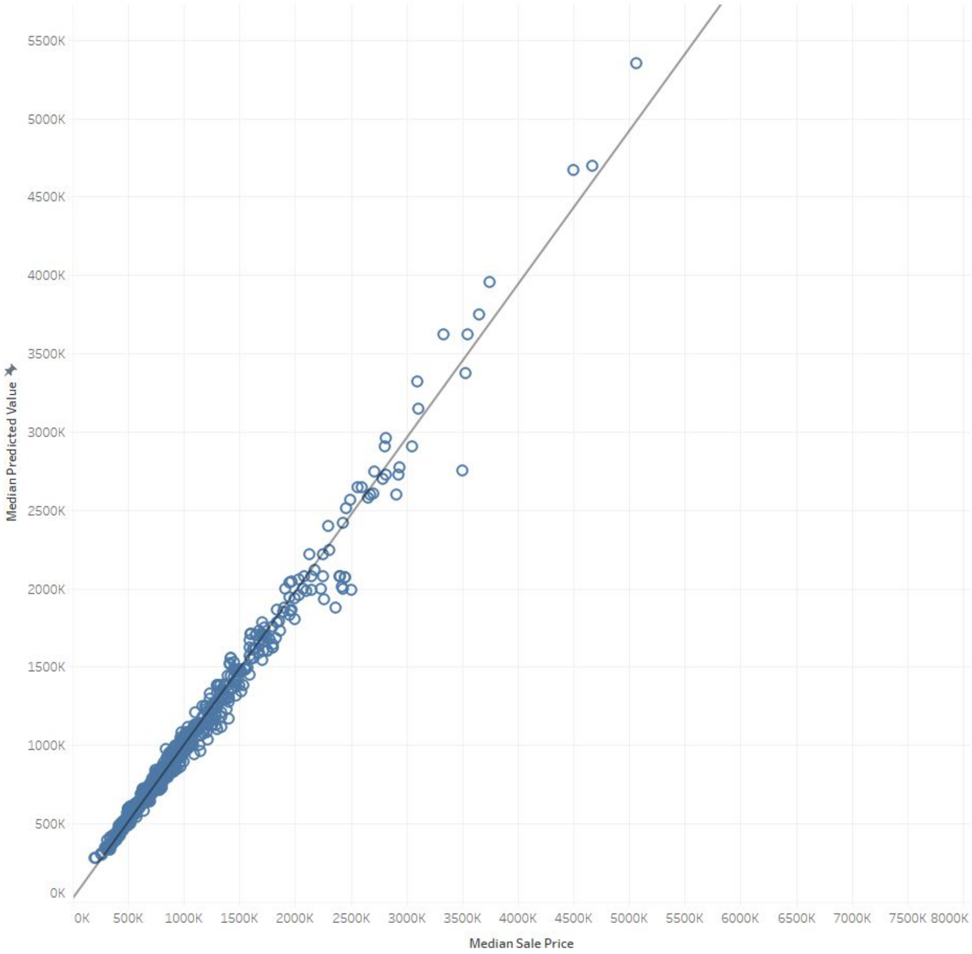
HOUSING ANALYSIS DRILL DOWN

- This Dashboard present :
  - Days on Market Mapping : Avg. Number of Days from Publish to Close .
  - Sale Distribution Per SQF: Distribution of House sales transactions per SQF.
  - Sale Distribution per # of rooms shows the distribution of House sales transactions per # of rooms.



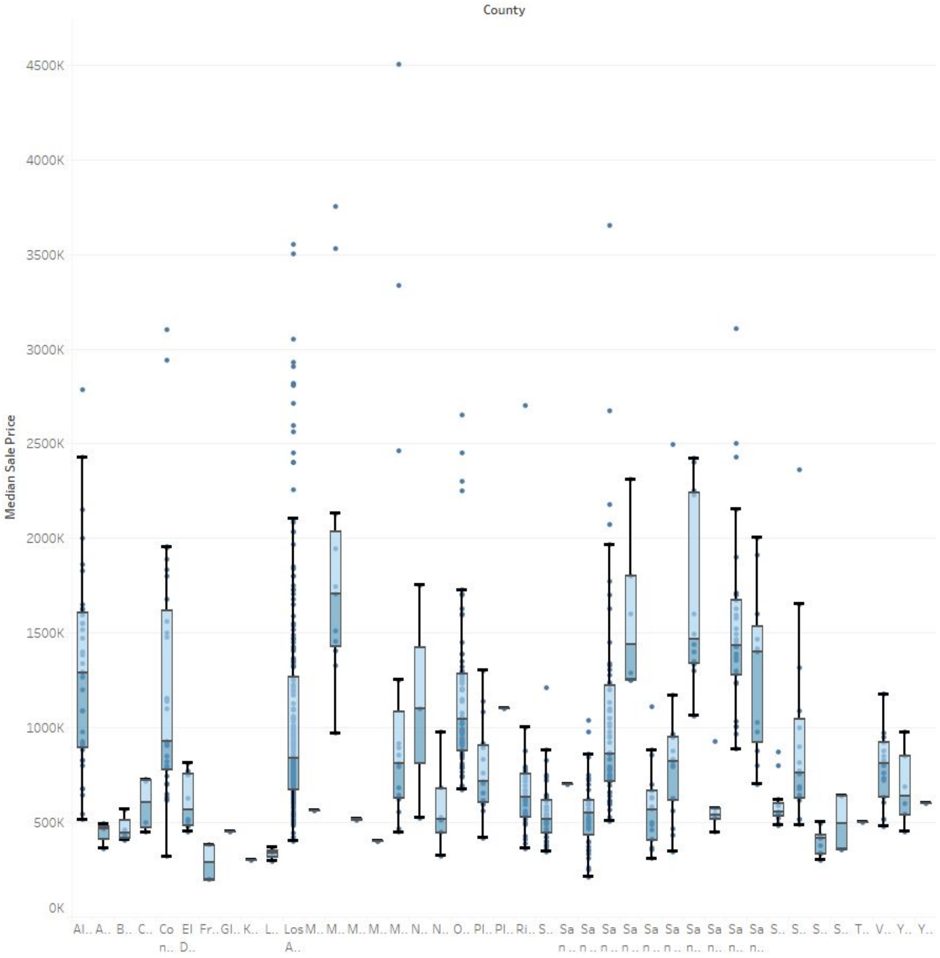
Additional Sources

Predicted Price Vs. Sale Price Regression Analysis



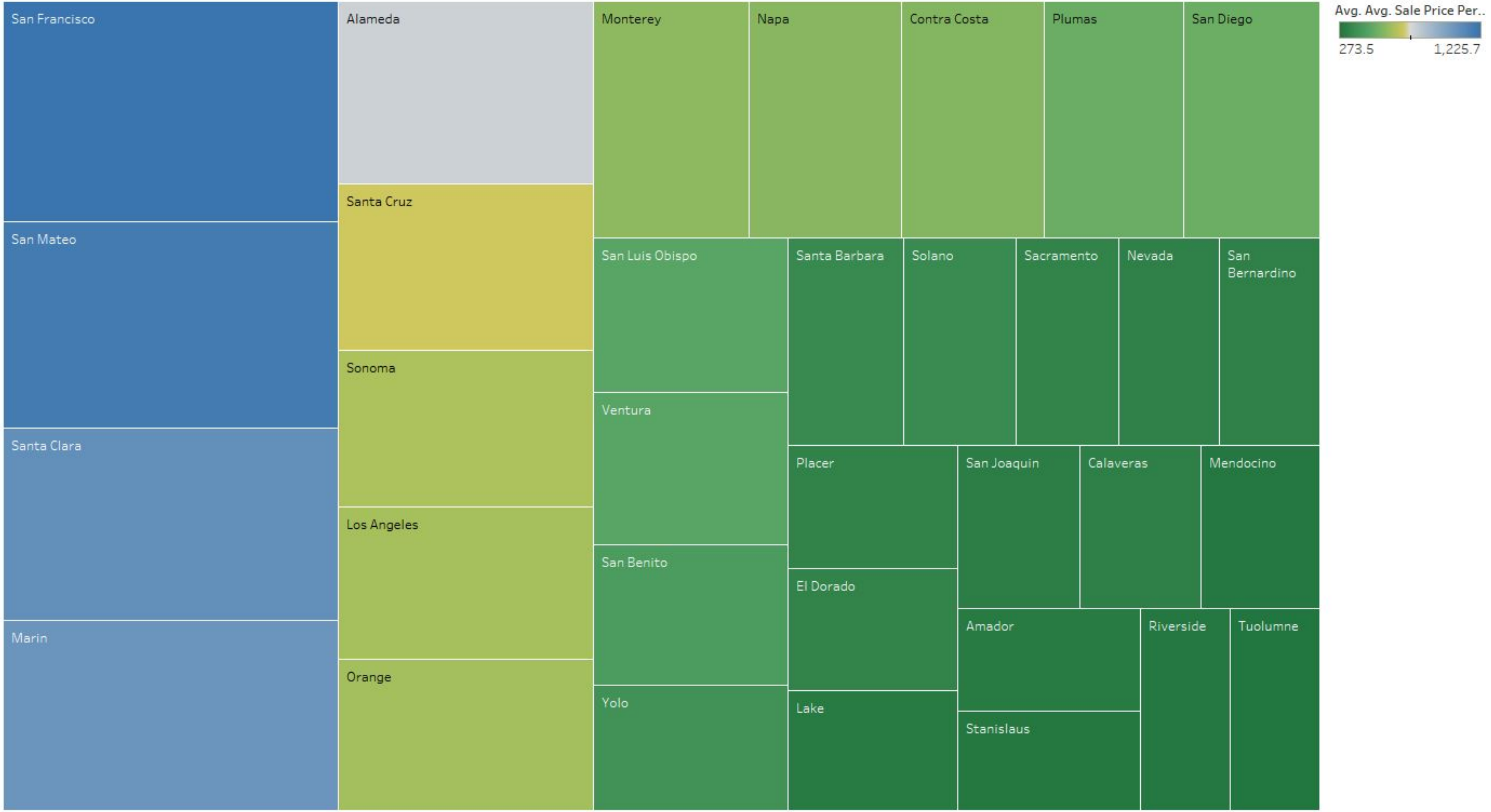
Median of Sale Price vs. median of Predicted Value. Details are shown for Zip Code. The data is filtered on County, average of Avg. Sale Price Per SQF and Sale Date. The County filter keeps 46 of 46 members. The average of Avg. Sale Price Per SQF filter ranges from 267 to 1,225.734066032. The Sale Date filter ranges from 5/31/2020 to 5/30/2021 and keeps Null values. The view is filtered on median of Sale Price and Zip Code. The median of Sale Price filter ranges from 92,000 to 23,050,000. The Zip Code filter keeps 941 of 941 members.

Median Sale Price Wisker Plot Per County



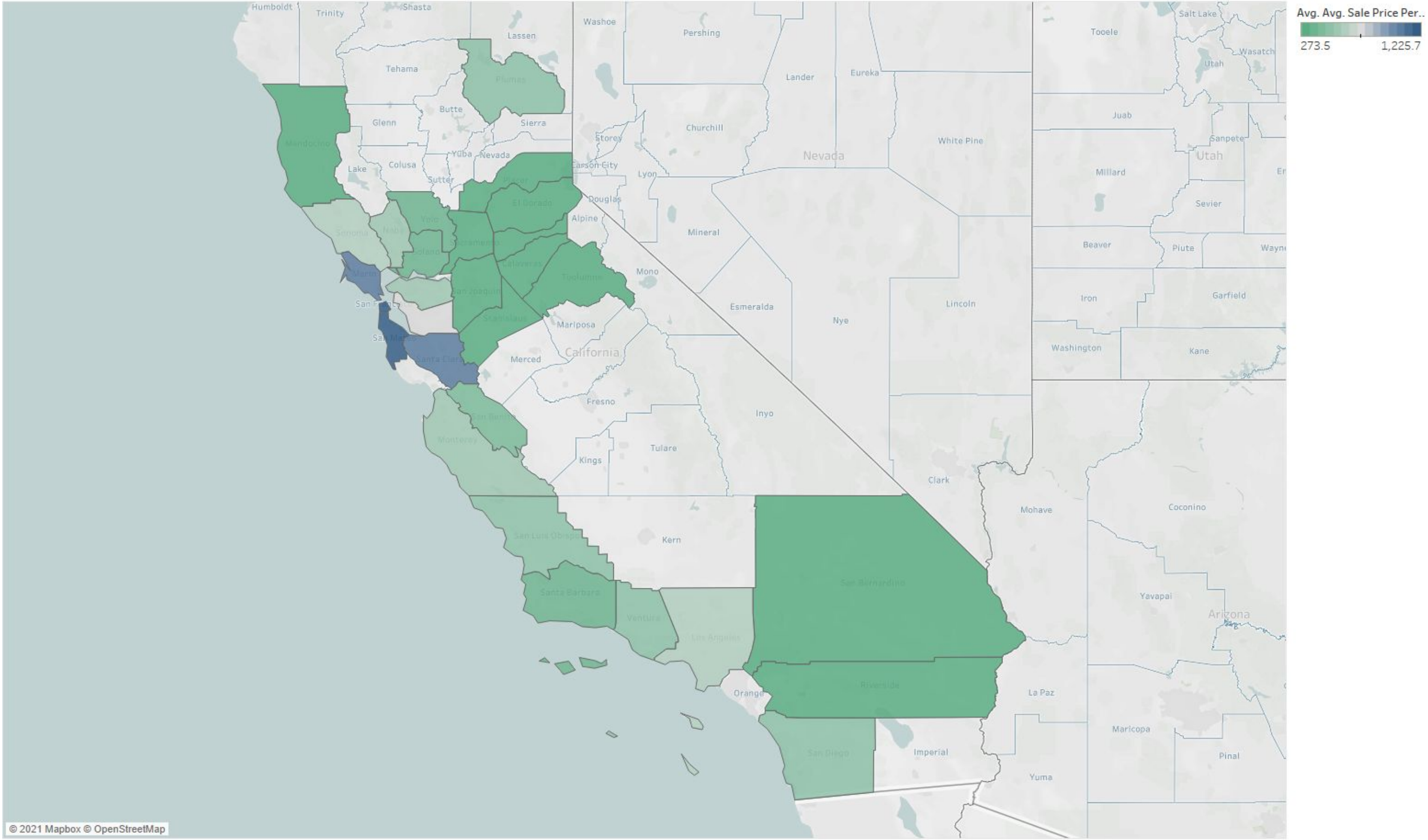
Median of Sale Price for each County. Details are shown for Zip Code. The data is filtered on average of Avg. Sale Price Per SQF and Sale Date. The average of Avg. Sale Price Per SQF filter ranges from 267 to 1,225.734066032. The Sale Date filter ranges from 5/31/2020 to 5/30/2021 and keeps Null values. The view is filtered on Exclusions (County,Zip Code), County and Zip Code. The Exclusions (County,Zip Code) filter keeps 935 members. The County filter keeps 46 of 46 members. The Zip Code filter keeps 941 of 941 members.

Heat Map: Avg. House Price, Prediction vs. Sale per County



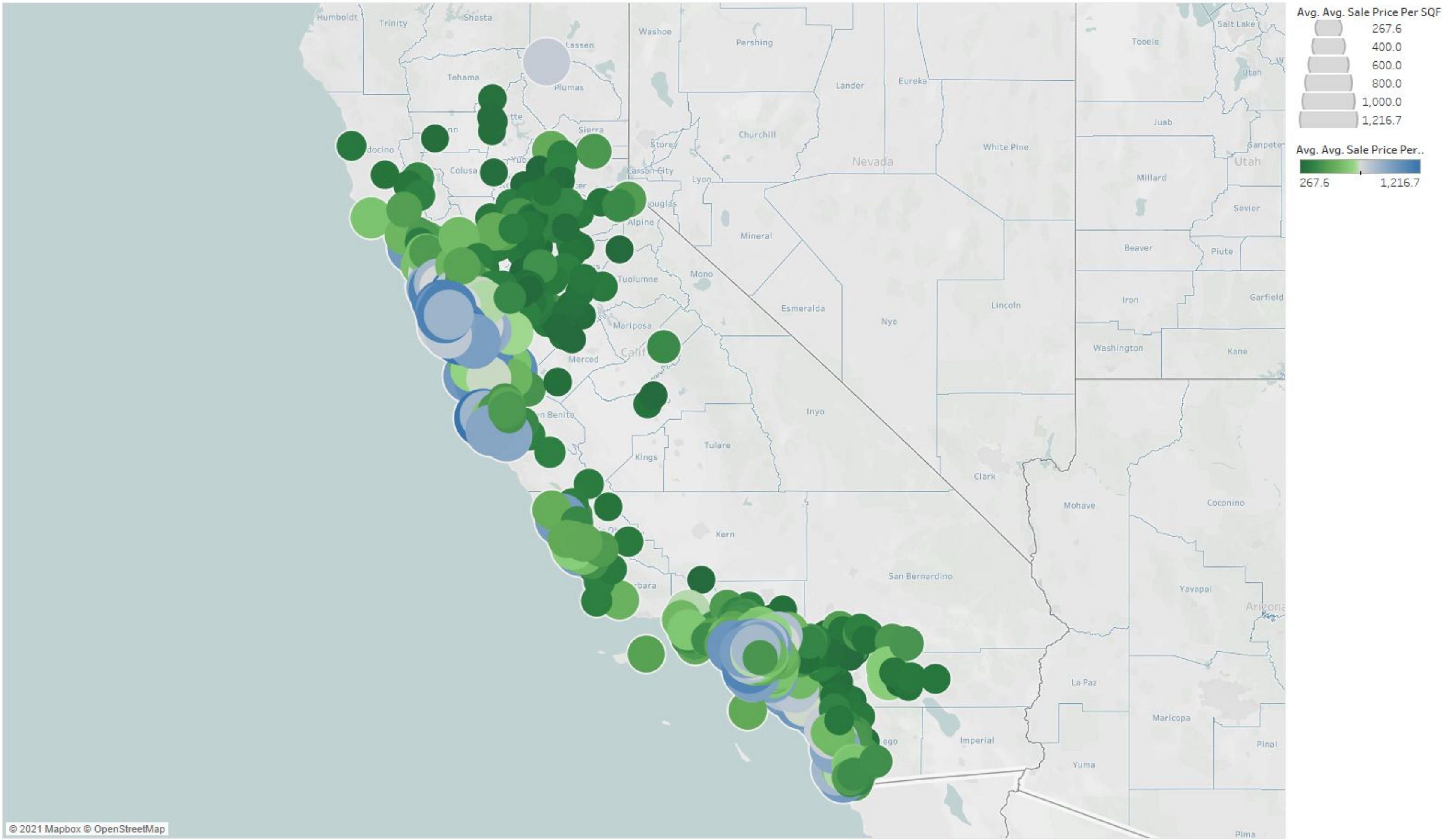
County. Color shows average of Avg. Sale Price Per SQF. Size shows average of Avg. Predicted Price Per SQF. The marks are labeled by County. The data is filtered on Sale Date and Zip Code. The Sale Date filter ranges from 5/31/2020 to 5/30/2021 and keeps Null values. The Zip Code filter keeps 941 of 941 members. The view is filtered on average of Avg. Sale Price Per SQF, average of Avg. Predicted Price Per SQF and County. The average of Avg. Sale Price Per SQF filter ranges from 267.0 to 1,225.7. The average of Avg. Predicted Price Per SQF filter ranges from 202.0 to 1,208.5. The County filter keeps 46 of 46 members.

Heat Map for : Avg. Price Prediction vs. Sale per SQF per County



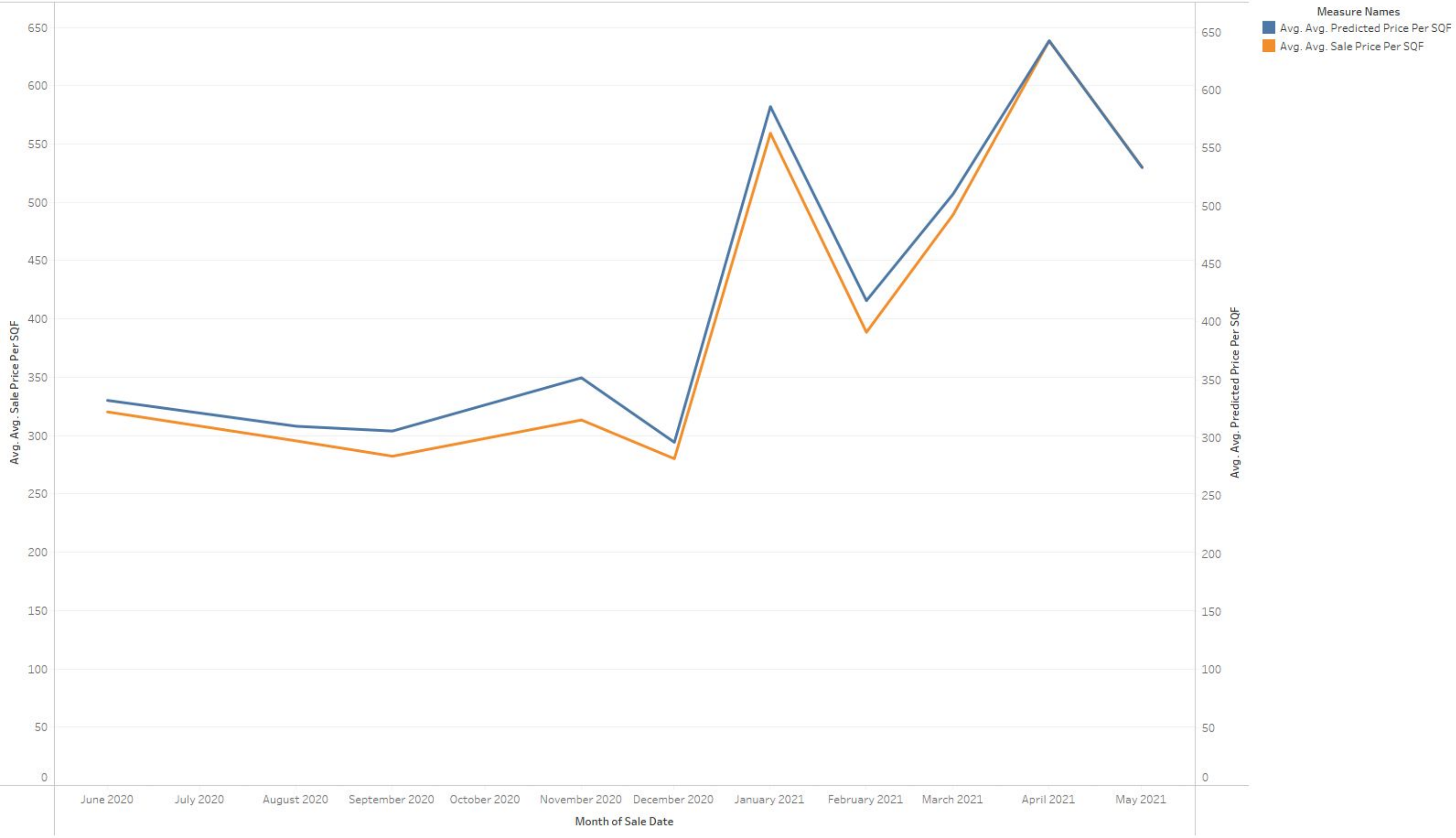
Map based on Longitude (generated) and Latitude (generated). Color shows average of Avg. Sale Price Per SQF. Details are shown for County. The data is filtered on Zip Code and Sale Date. The Zip Code filter keeps 941 of 941 members. The Sale Date filter ranges from 5/31/2020 to 5/30/2021 and keeps Null values. The view is filtered on County and average of Avg. Sale Price Per SQF. The County filter keeps 46 of 46 members. The average of Avg. Sale Price Per SQF filter ranges from 267.0 to 1,225.7.

Avg. House Price, Prediction vs. Sale per County



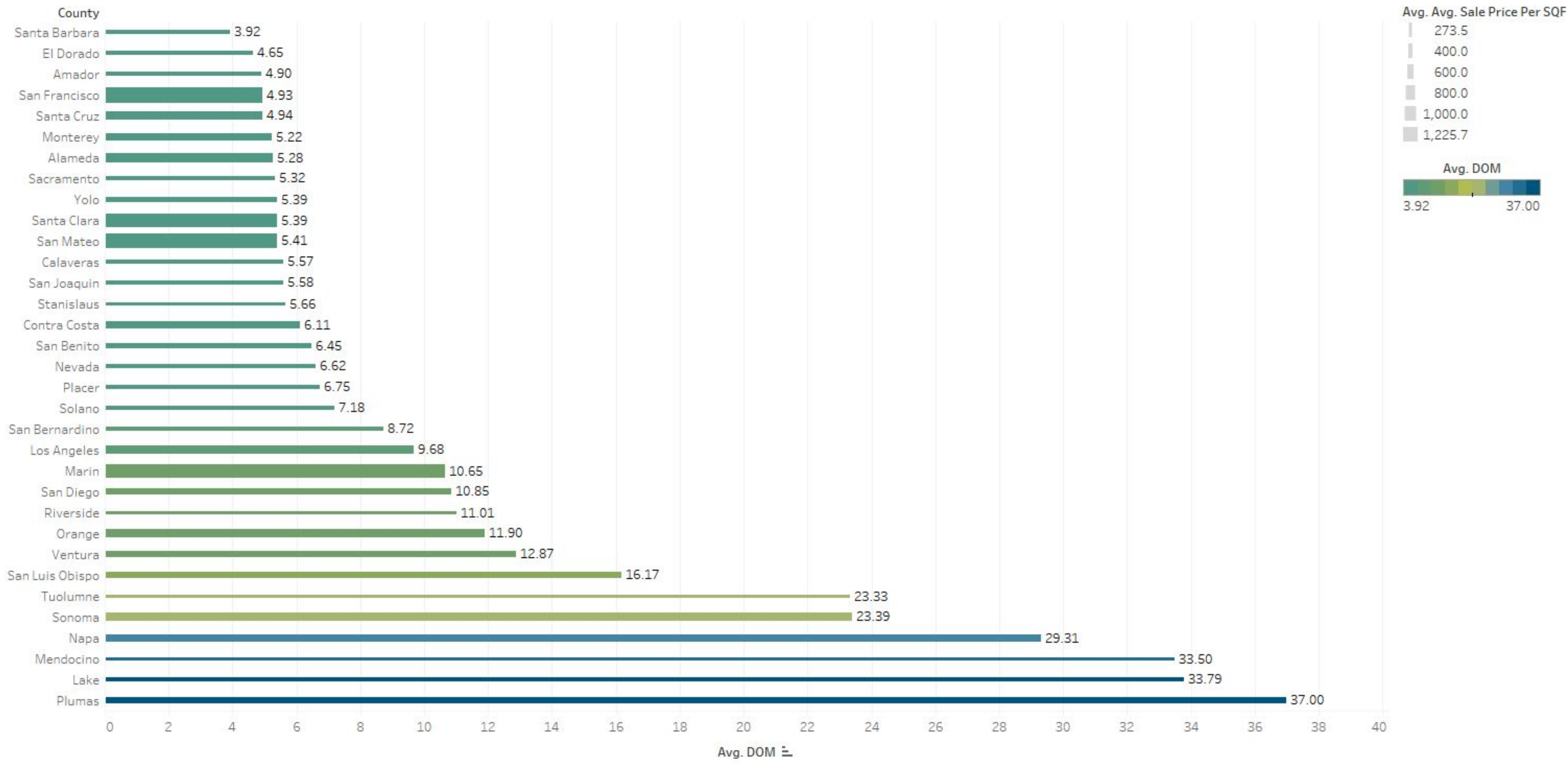
Map based on Longitude (generated) and Latitude (generated). Color shows average of Avg. Sale Price Per SQF. Size shows average of Avg. Sale Price Per SQF. Details are shown for Zip Code. The data is filtered on County and Sale Date. The County filter keeps 46 of 46 members. The Sale Date filter ranges from 5/31/2020 to 5/30/2021 and keeps Null values. The view is filtered on Zip Code and average of Avg. Sale Price Per SQF. The Zip Code filter keeps 941 of 941 members. The average of Avg. Sale Price Per SQF filter ranges from 267.0 to 1,225.7.

CA Housing: Avg. Predicted Prices vs. Avg. Sale Price



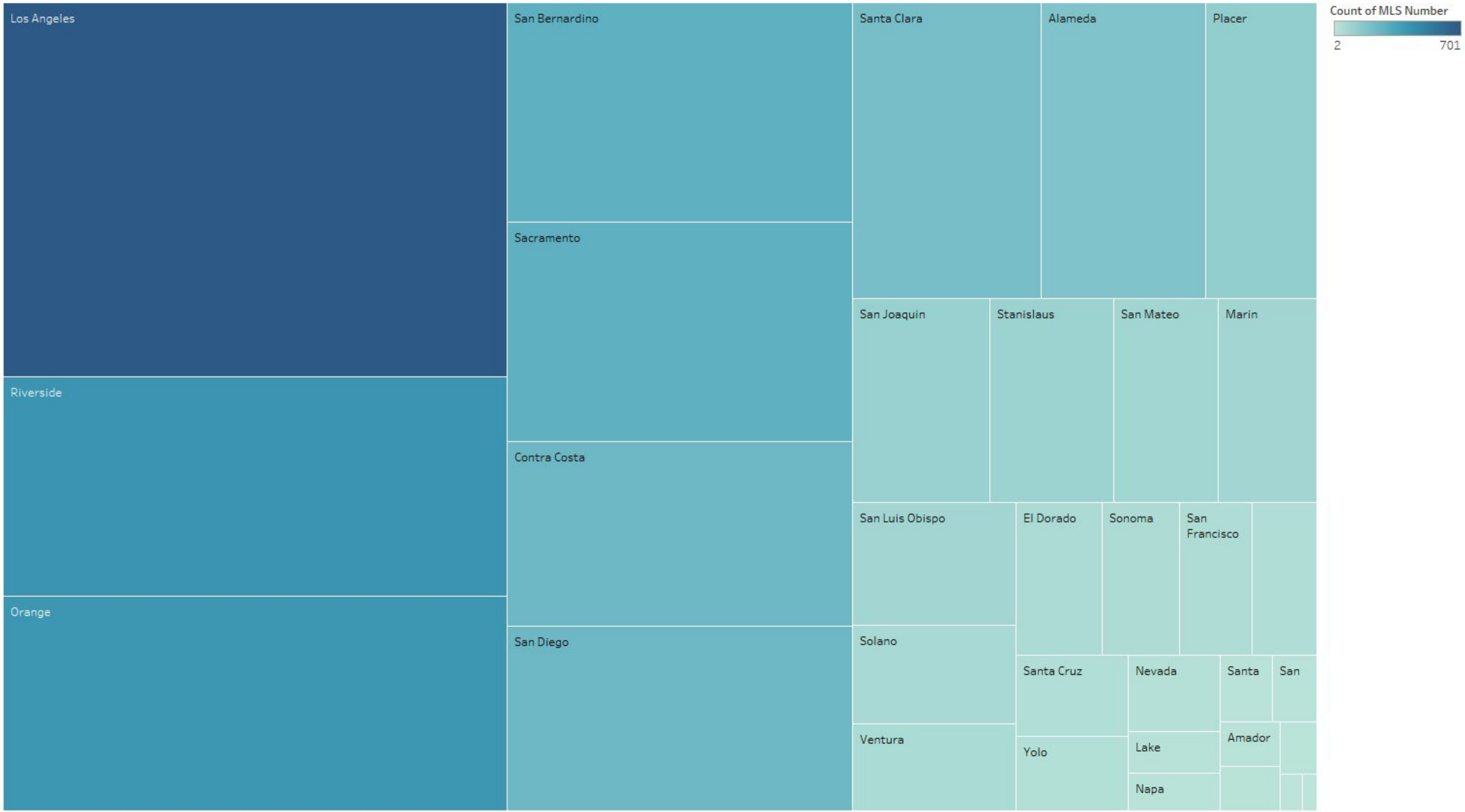
The trends of Avg. Avg. Sale Price Per SQF and Avg. Avg. Predicted Price Per SQF for Sale Date Month. Color shows details about Avg. Avg. Sale Price Per SQF and Avg. Avg. Predicted Price Per SQF. The data is filtered on County, Zip Code, Sale Date Year, Sale Date Quarter and Sale Date. The County filter keeps 46 of 46 members. The Zip Code filter keeps 941 of 941 members. The Sale Date Year filter keeps 2020 and 2021. The Sale Date Quarter filter has multiple members selected. The Sale Date filter ranges from 5/31/2020 to 5/30/2021 and keeps Null values. The view is filtered on average of Avg. Sale Price Per SQF, which ranges from 267.0 to 1,225.7.

Days On Market Mapping



Average of DOM for each County. Color shows average of DOM. Size shows average of Avg. Sale Price Per SQF. The marks are labeled by average of DOM. Details are shown for County. The data is filtered on Zip Code and Sale Date. The Zip Code filter keeps 941 of 941 members. The Sale Date filter ranges from 5/31/2020 to 5/30/2021 and keeps Null values. The view is filtered on average of DOM, County and average of Avg. Sale Price Per SQF. The average of DOM filter includes everything. The County filter keeps 46 of 46 members. The average of Avg. Sale Price Per SQF filter ranges from 267.0 to 1,225.7.

Number of Sales records per County

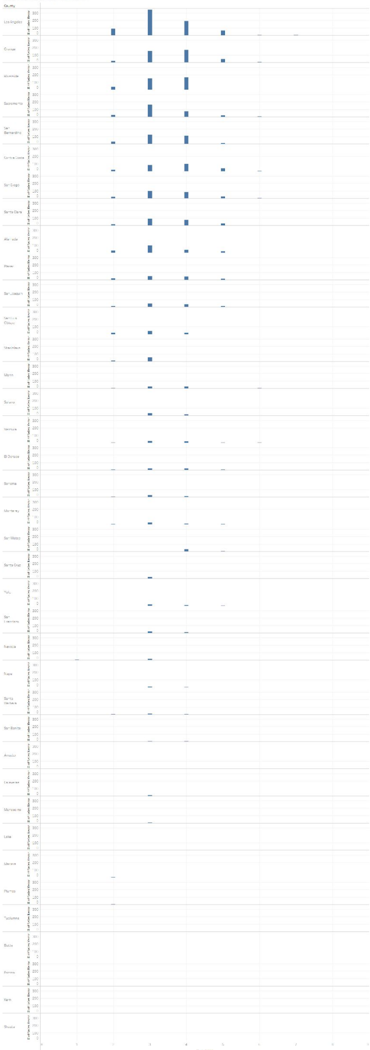


County. Color shows count of MLS Number. Size shows count of MLS Number. The marks are labeled by County. The data is filtered on Zip Code, average of Avg. Sale Price Per SQF and Sale Date. The Zip Code filter keeps 941 of 941 members. The average of Avg. Sale Price Per SQF filter ranges from 267 to 1,225.734066032. The Sale Date filter ranges from 5/31/2020 to 5/30/2021 and keeps Null values. The view is filtered on County, which keeps 46 of 46 members.

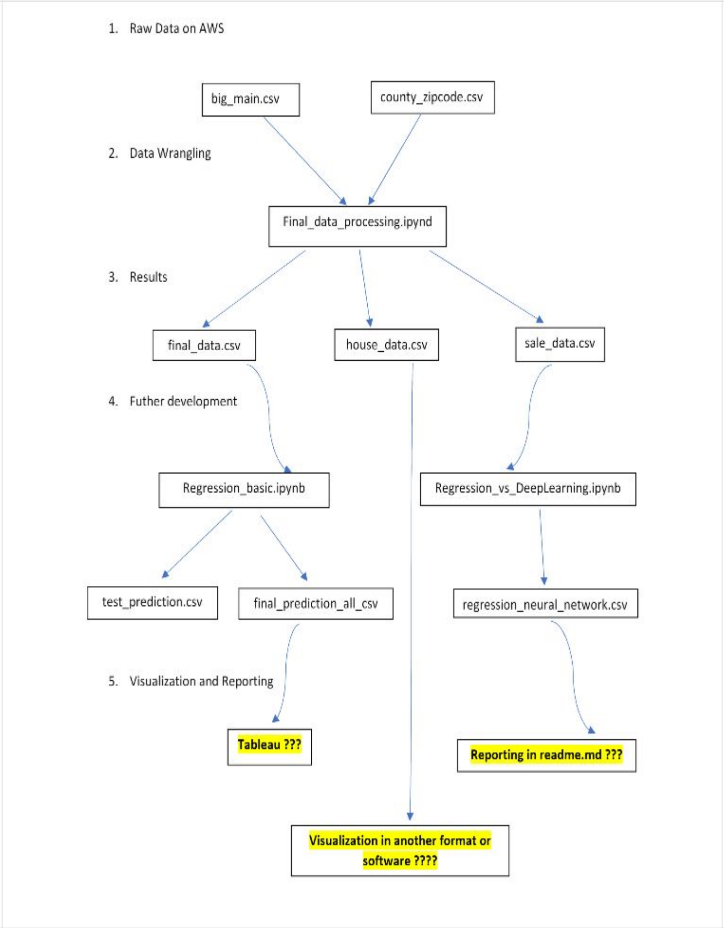
Sale Distribution per SQF

The trend of count of Zip Code for Sq Ft Total (bin) broken down by County. The data is filtered on Sq Ft Total, Zip Code, average of Avg. Sale Price Per SQF and Sale Date. The Sq Ft Total filter ranges from 157 to 13028 and keeps Null values. The Zip Code filter keeps 941 of 941 members. The average of Avg. Sale Price Per SQF filter ranges from 237 to 1,225.734056032. The Sale Date filter ranges from 5/31/2020 to 5/30/2023 and keeps Null values. The view is filtered on County and Sq Ft Total (bin). The County filter keeps 46 of 46 members. The Sq Ft Total (bin) filter ranges from 203 to 13013.

Sale Distribution per # of Rooms



The plot of count of Zip Code for Beds Total broken down by County. The data is filtered on Zip Code, average of Avg. Sale Price Per SQF and Sale Date. The Zip Code filter keeps 941 of 941 members. The average of Avg. Sale Price Per SQF filter ranges from 267 to 1,235.734066032. The Sale Date filter ranges from 6/31/2020 to 6/30/2021 and keeps Null values. The view is filtered on County and count of Zip Code. The County filter keeps 46 of 46 members. The count of Zip Code filter ranges from 1 to 346.


1. Raw Data on AWS
big_main.csv
county_zipcode.csv
2. Data Wrangling
Final_data_processing.ipynd
3. Results
final_data.csv
house_data.csv
sale_data.csv
4. Futher development
Regression_basic.ipynb
Regression_vs_DeepLearning.ipynb
test_prediction.csv
final_prediction_all_csv
regression_neural_network.csv
5. Visualization and Reporting
Tableau ???
Reporting in readme.md ???
Visualization in another format or software ????

Connection, Regression, Neural network, Random Forest

TrongQuyen Nguyen

```
SUMMARY OF R2_SCORE

Model 1: Regression with List_Price: 0.9605332854433306

Model 2: Regression withOUT List_Price: 0.672214220079839З

Model 3: Neural Network with List_Price: 0.951540873132З525

Model 4: Neural Network withOUT List_Price: 0.44249087024205247

Model 5: Random Forest with List_Price: 0.9650218426634006

Model 6: Random Forest withOUT List_Price: 0.64523957894399
```

With “List_Price” R2_score is the higher than without