Mark Ehler

King County Housing Data

About this Project

The task:

Given past house sales, predict what properties will be sell for high values when they come on the market.

The data:

21,597 samples of housing sales from May, 2014 until May, 2015 in the King County area.

About the Data

---

20 individual data points for each sample

6 statistics on square footage

3 statistics on geographical location

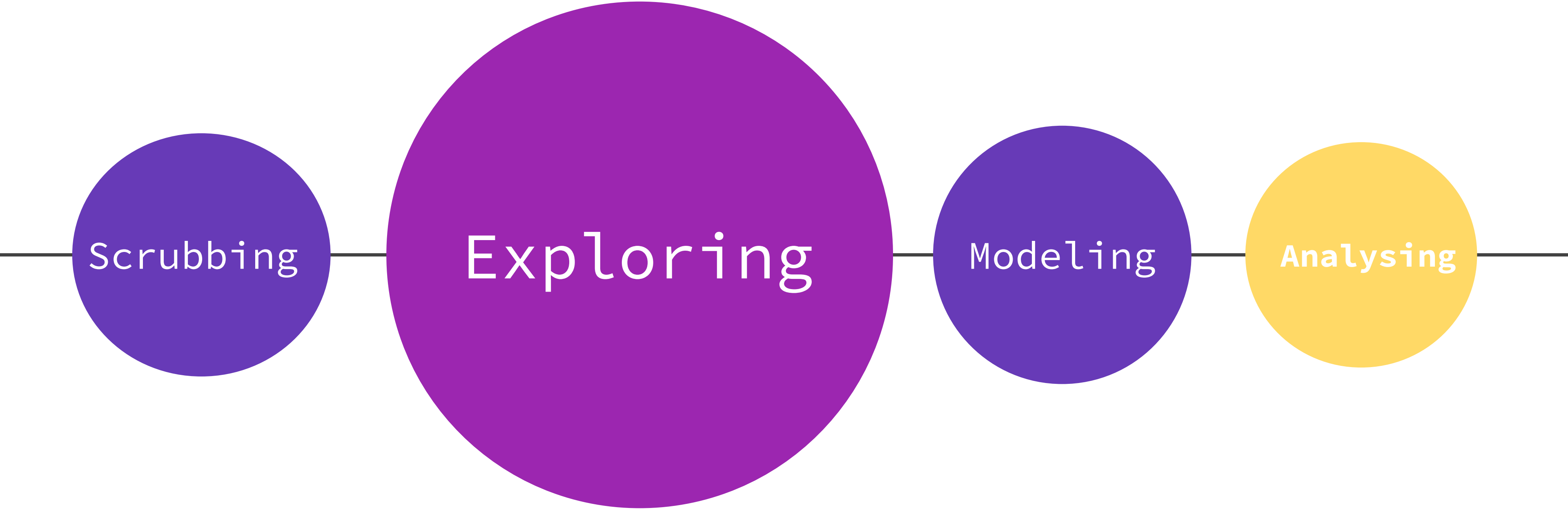
3 statistics dealing with dates

3 strictly quantifiable categories

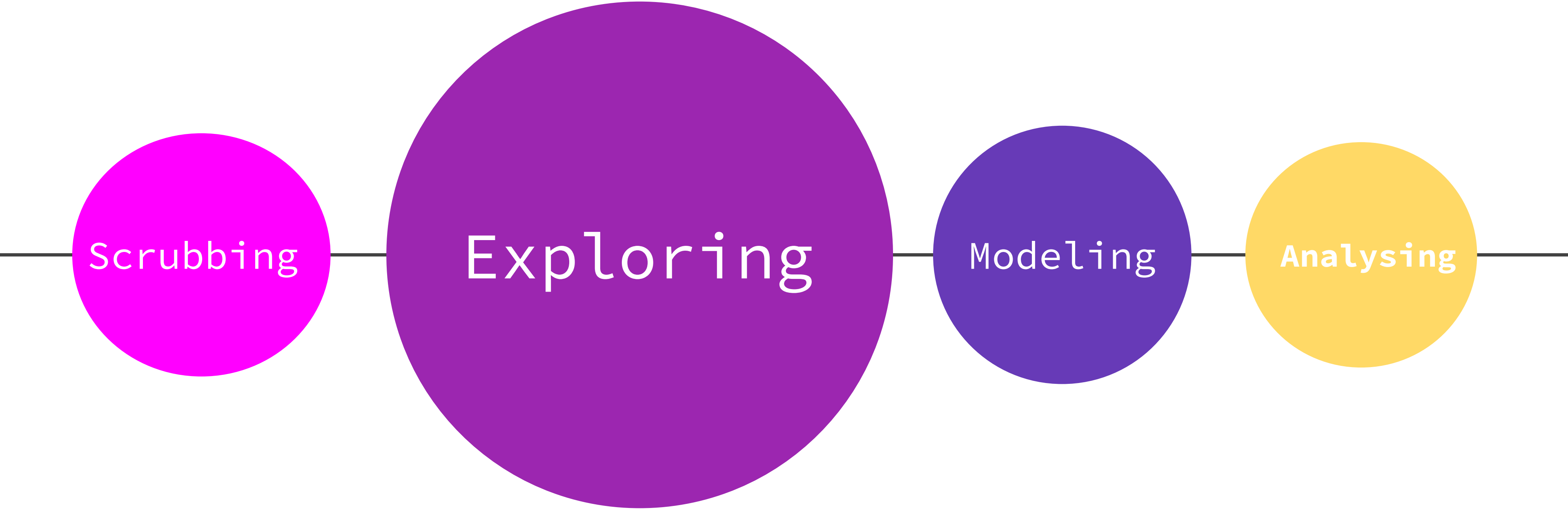
4 subjective values given by the researchers

1 ID tag

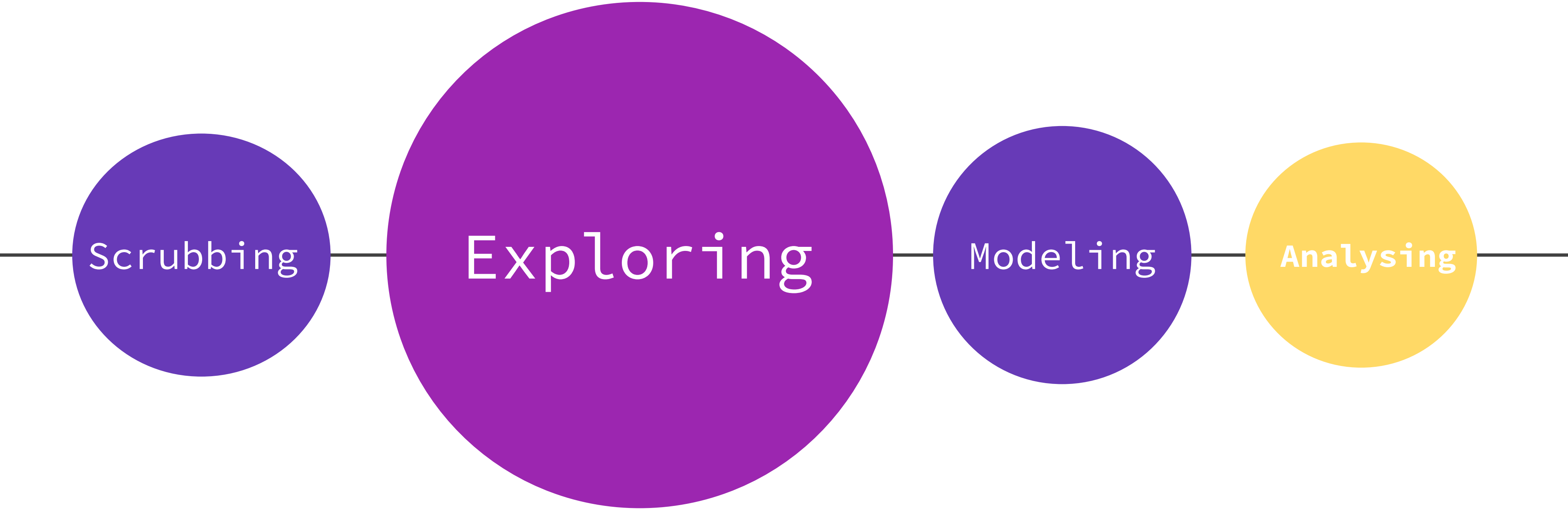
The Process

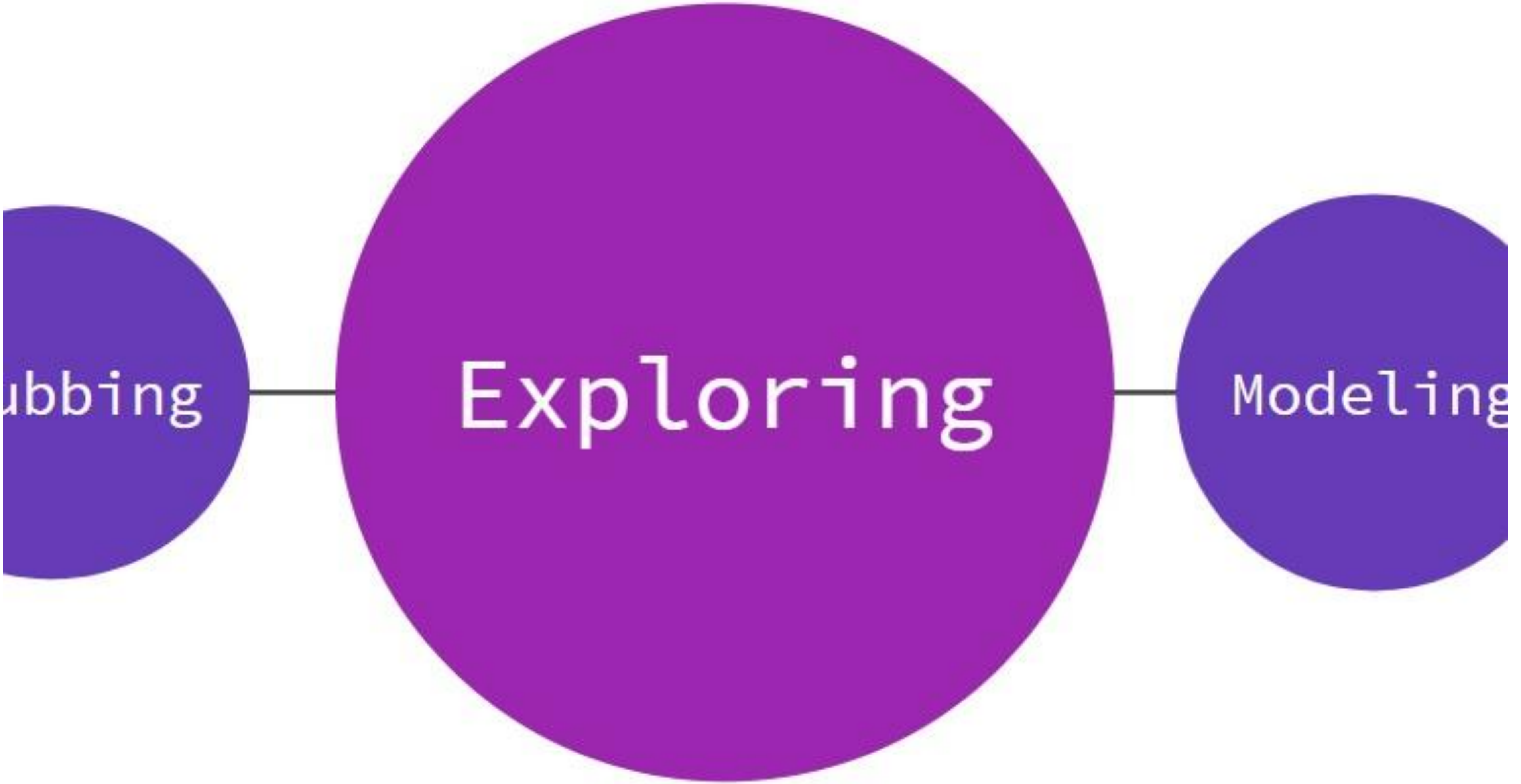


The Process

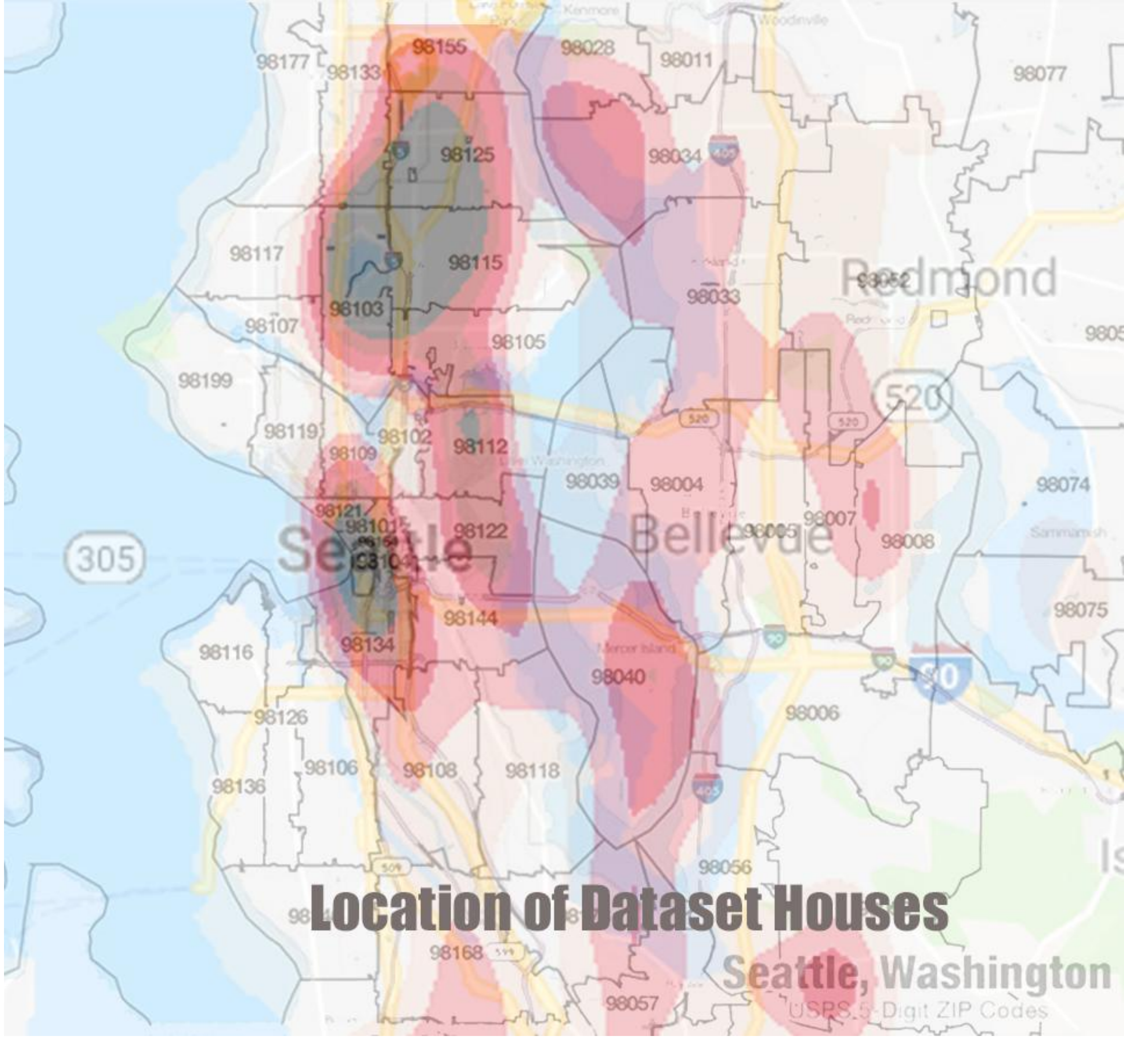


The Process

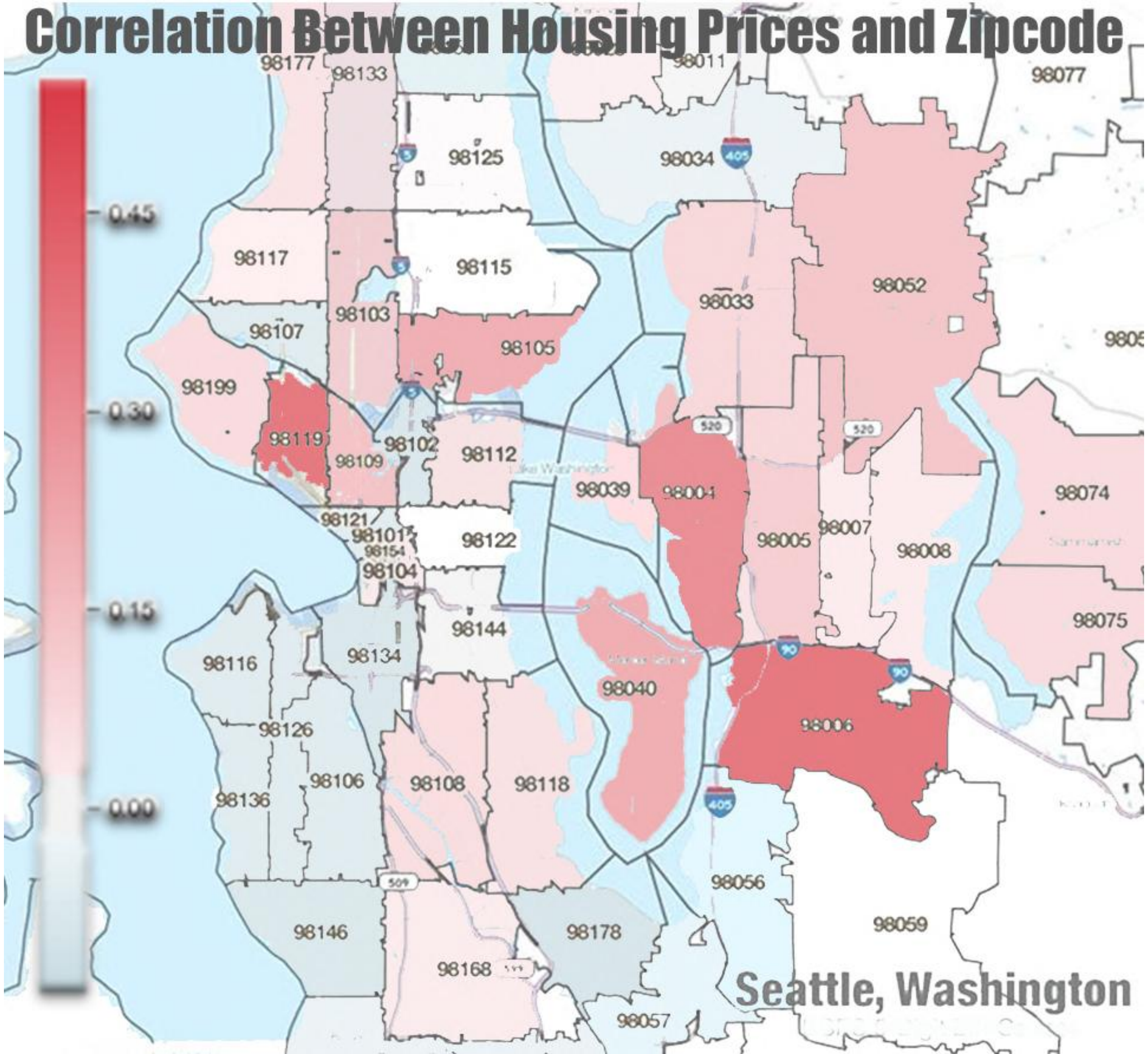



ubbing
Exploring
Modeling

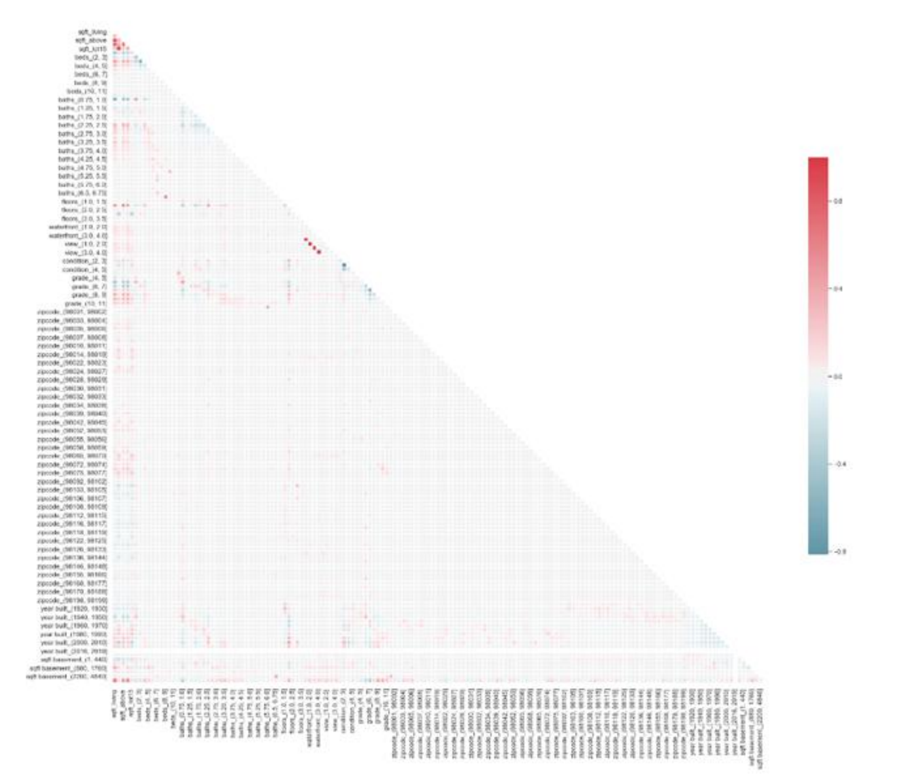
King County

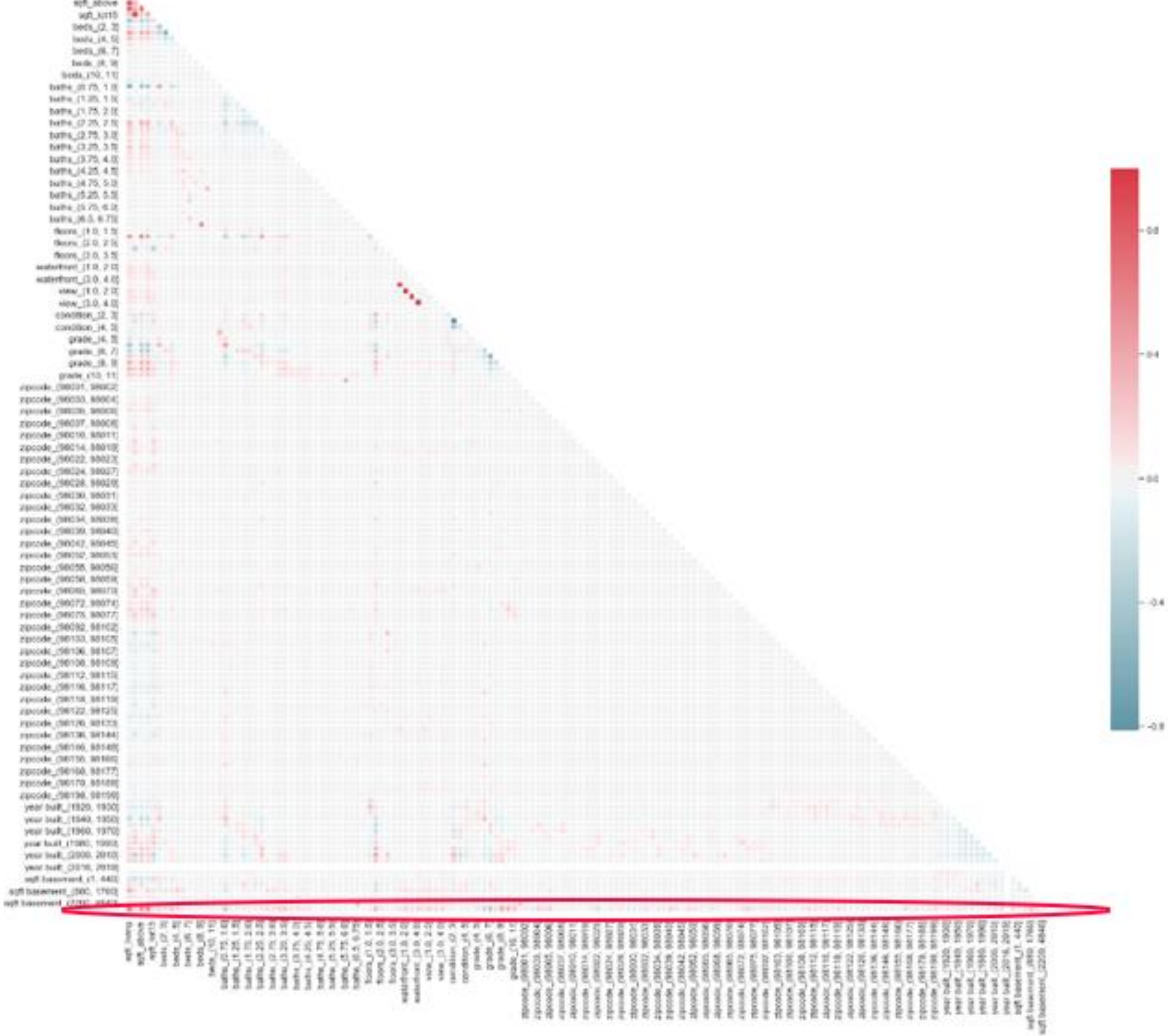

Location of Dataset Houses
Seattle, Washington
USPS 5-Digit ZIP Codes
Seattle
Bellevue
Redmond

Price related to Zipcode

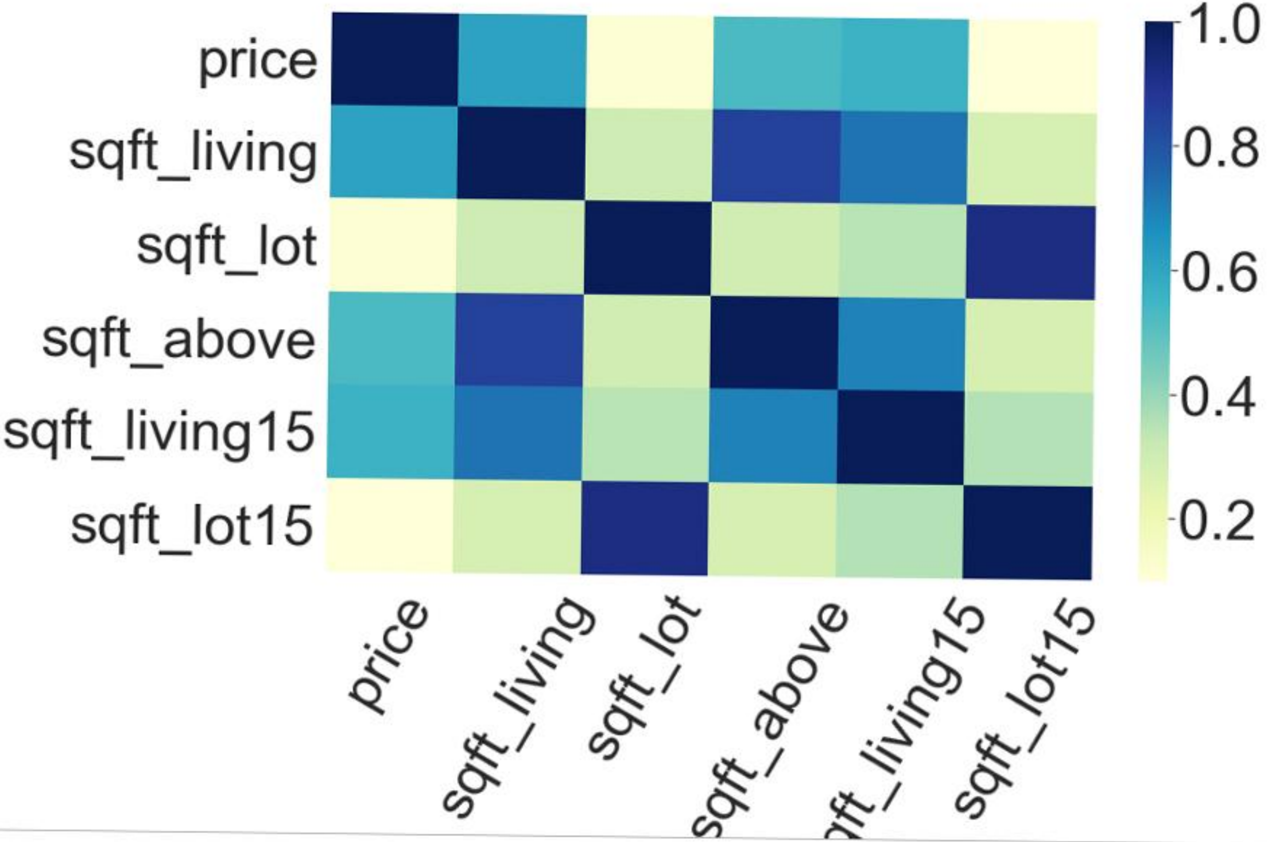


Power Overwhelming!





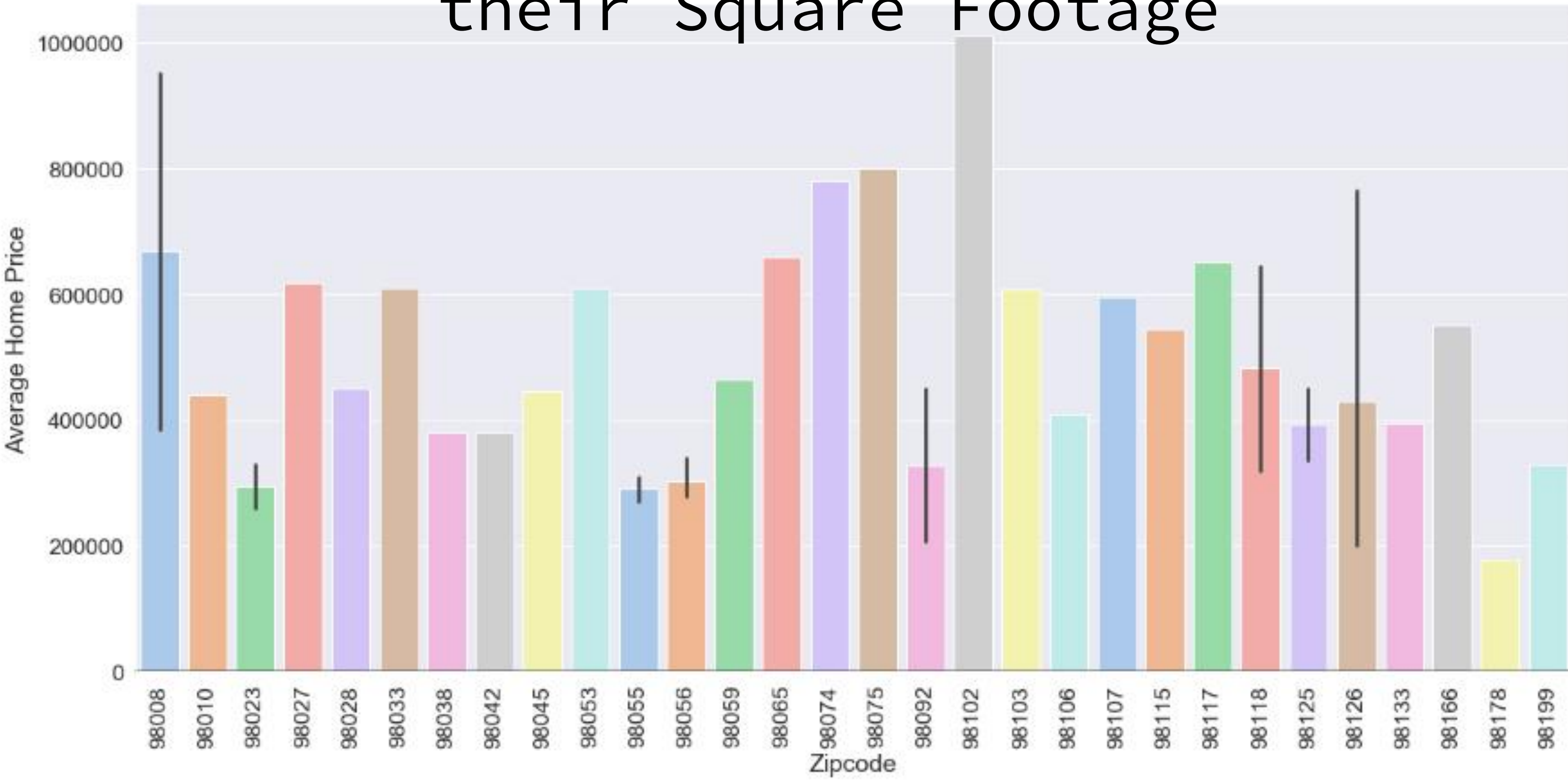
Correlation Brtween Continuous Variables of our Data



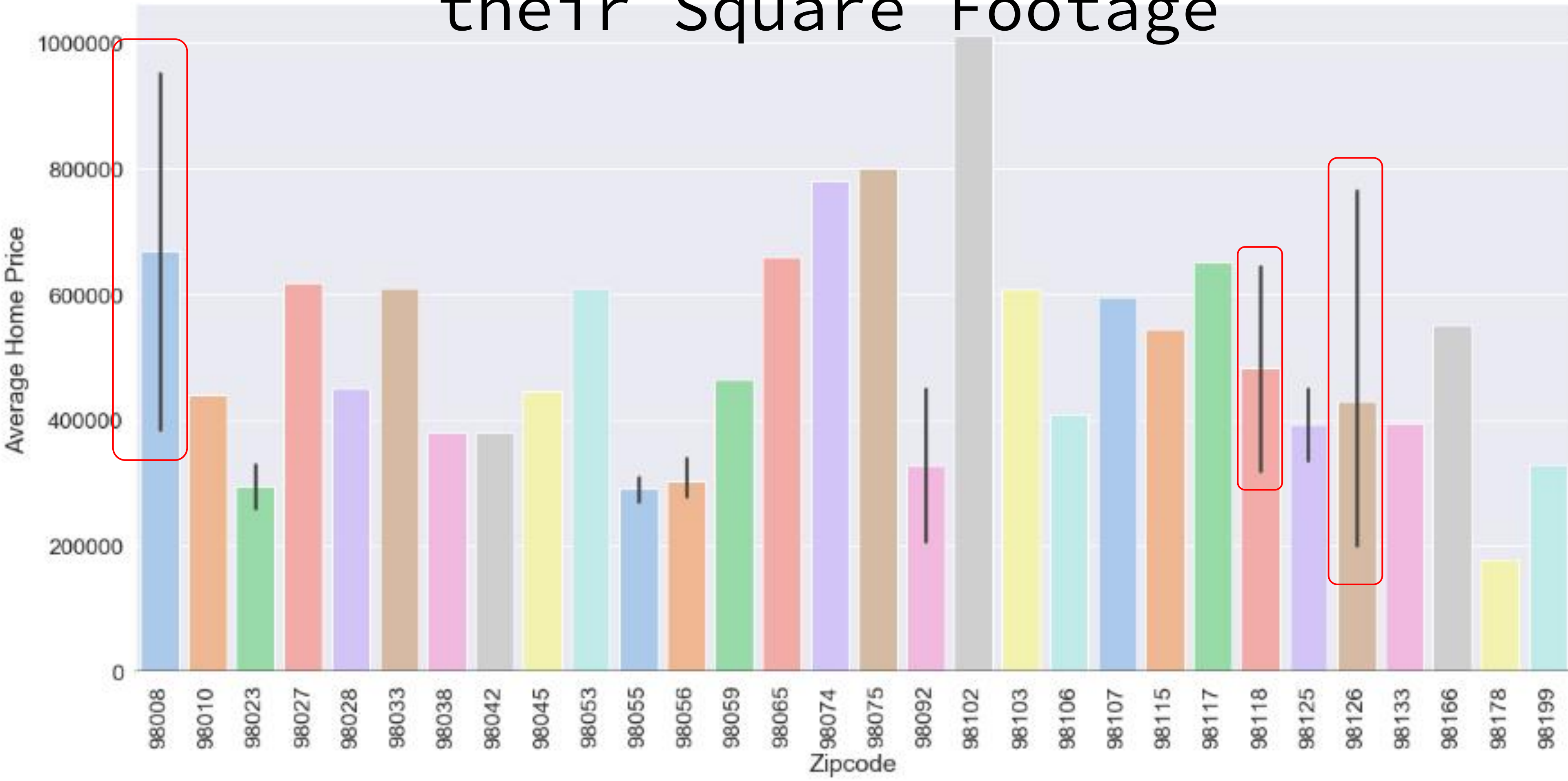
Samples of Home Prices and Their Square Footage

Samples of Home Prices and Their Square Footage

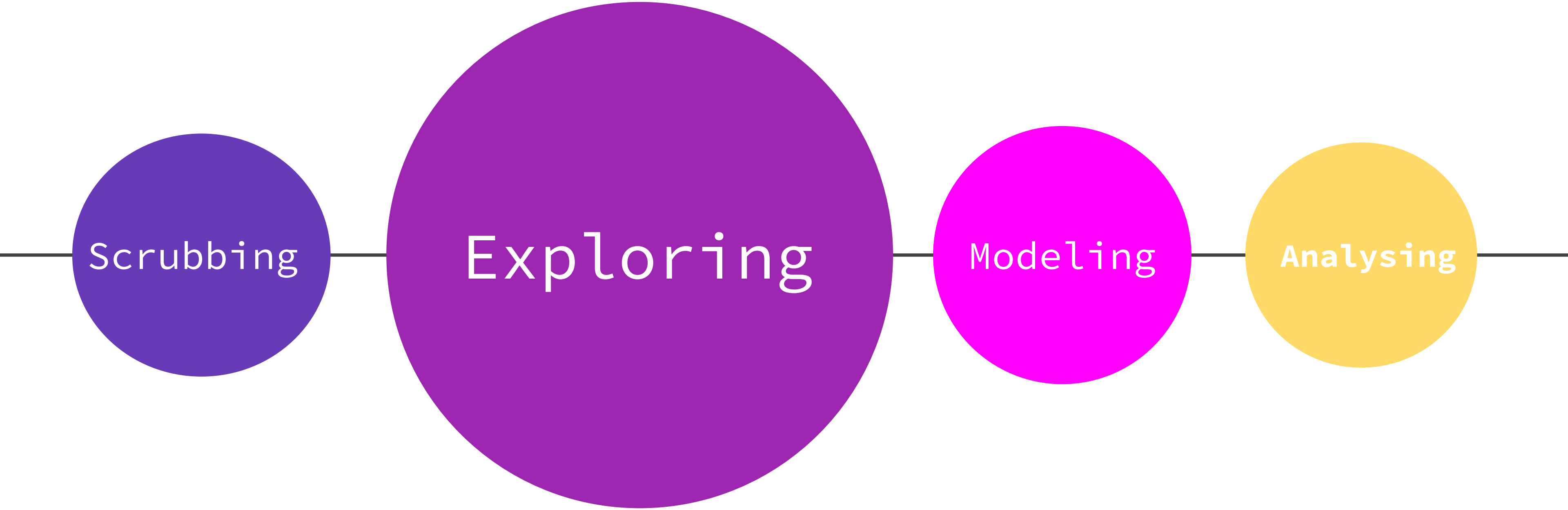
Samples of Zip Codes and their Square Footage



Samples of Zip Codes and their Square Footage



The Process



OLS Regression Results

In [130]:
```
model.summary()
```

Out[130]:

OLS Regression Results

Dep. Variable:	price	**R-squared:**	0.844
Model:	OLS	**Adj. R-squared:**	0.843
Method:	Least Squares	**F-statistic:**	754.4
Date:	Fri, 25 Jan 2019	**Prob (F-statistic):**	0.00
Time:	18:19:05	**Log-Likelihood:**	-2.6590e+05
No. Observations:	20756	**AIC:**	5.321e+05
Df Residuals:	20607	**BIC:**	5.333e+05
Df Model:	148		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	-1.439e+05	1.02e+05	-1.411	0.158	-3.44e+05	5.61e+04
sqft_living	1.866e+05	2.73e+04	6.833	0.000	1.33e+05	2.4e+05
sqft_lot	2.767e+05	1.5e+04	18.418	0.000	2.47e+05	3.06e+05
sqft_above	3.797e+05	2.53e+04	15.004	0.000	3.3e+05	4.29e+05
sqft_living15	1.35e+05	9319.504	14.490	0.000	1.17e+05	1.53e+05
sqft_lot15	-5.82e+04	1.47e+04	-3.968	0.000	-8.7e+04	-2.95e+04
beds_(1, 2]	-8281.9848	6901.719	-1.200	0.230	-2.18e+04	5245.931
beds_(2, 3]	-1.088e+04	6984.881	-1.557	0.119	-2.46e+04	2812.882
beds_(3, 4]	-1.119e+04	7186.644	-1.557	0.120	-2.53e+04	2898.758
beds_(4, 5]	-2.06e+04	7612.035	-2.707	0.007	-3.55e+04	-5683.931
beds_(5, 6]	-2.711e+04	9431.195	-2.874	0.004	-4.56e+04	-8619.441
beds_(6, 7]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]

OLS Regression Results

In [130]:
```
model.summary()
```

Out[130]:

OLS Regression Results

Dep. Variable:	price	**R-squared:**	0.844
Model:	OLS	**Adj. R-squared:**	0.843
Method:	Least Squares	**F-statistic:**	754.4
Date:	Fri, 25 Jan 2019	**Prob (F-statistic):**	0.00
Time:	18:19:05	**Log-Likelihood:**	-2.6590e+05
No. Observations:	20756	**AIC:**	5.321e+05
Df Residuals:	20607	**BIC:**	5.333e+05
Df Model:	148		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	-1.439e+05	1.02e+05	-1.411	0.158	-3.44e+05	5.61e+04
sqft_living	1.866e+05	2.73e+04	6.833	0.000	1.33e+05	2.4e+05
sqft_lot	2.767e+05	1.5e+04	18.418	0.000	2.47e+05	3.06e+05
sqft_above	3.797e+05	2.53e+04	15.004	0.000	3.3e+05	4.29e+05
sqft_living15	1.35e+05	9319.504	14.490	0.000	1.17e+05	1.53e+05
sqft_lot15	-5.82e+04	1.47e+04	-3.968	0.000	-8.7e+04	-2.95e+04
beds_(1, 2]	-8281.9848	6901.719	-1.200	0.230	-2.18e+04	5245.931
beds_(2, 3]	-1.088e+04	6984.881	-1.557	0.119	-2.46e+04	2812.882
beds_(3, 4]	-1.119e+04	7186.644	-1.557	0.120	-2.53e+04	2898.758
beds_(4, 5]	-2.06e+04	7612.035	-2.707	0.007	-3.55e+04	-5683.931
beds_(5, 6]	-2.711e+04	9431.195	-2.874	0.004	-4.56e+04	-8619.441
beds_(6, 7]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]

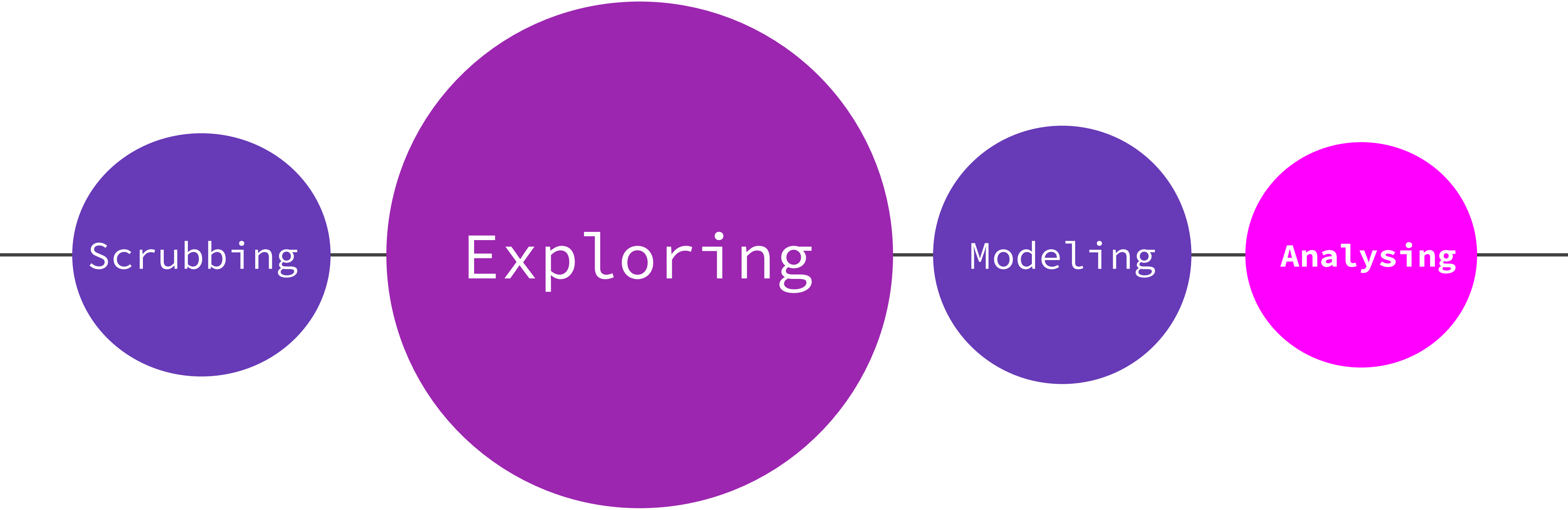
Smallest Predictive Errors

```
In [194]: top_ten = X.columns[selector.support_ == True]
          list(top_ten)

Out[194]: ['sqft_living',
           'sqft_living15',
           'baths_(5.75, 6.0]',
           'grade_(11, 12]',
           'zipcode_        98004]',
           'zipcode_        98039]',
           'zipcode_        98040]',
           'zipcode_        98109]',
           'zipcode_        98112]',
           'zipcode_        98119]']
```

Least Accurate Variables

```
In [266]: dropped = X.drop(returns, axis=1)
          dropped.columns

Out[266]: Index(['beds_(1, 2]', 'beds_(2, 3]', 'beds_(3, 4]', 'beds_(5, 6]',
                 'beds_(7, 8]', 'beds_(8, 9]', 'beds_(9, 10]', 'beds_(10, 11]',
                 'baths_(0.5, 0.75]', 'baths_(0.75, 1.0]', 'baths_(1.0, 1.25]',
                 'baths_(2.0, 2.25]', 'baths_(2.25, 2.5]', 'baths_(4.25, 4.5]',
                 'baths_(4.5, 4.75]', 'baths_(4.75, 5.0]', 'baths_(5.0, 5.25]',
                 'baths_(5.25, 5.5]', 'baths_(5.5, 5.75]', 'baths_(6.0, 6.5]',
                 'baths_(6.5, 6.75]', 'baths_(6.75, 7.5]', 'floors_(1.0, 1.5]',
                 'floors_(2.0, 2.5]', 'floors_(3.0, 3.5]', 'grade_(3, 4]',
                 'grade_(4, 5]', 'grade_(6, 7]', 'zipcode_(98034, 98038]',
                 'zipcode_(98053, 98055]', 'zipcode_(98056, 98058]',
                 'zipcode_(98166, 98168]', 'zipcode_(98178, 98188]',
                 'zipcode_(98188, 98198]', 'year built_(1950, 1960]',
                 'year built_(1960, 1970]', 'year built_(1980, 1990]',
                 'year built_(1990, 2000]', 'year built_(2016, 2019]',
                 'sqft basement_(1, 440]', 'sqft basement_(440, 880]',
                 'sqft basement_(880, 1760]', 'sqft basement_(1760, 2200]',
                 'sqft basement_(2200, 4840]'],
                dtype='object')
```

The Process



R Squared After Dropping Variables

In [173]:
```
y = y
X = stepwise_X


preds_int = sm.add_constant(X)
model = sm.OLS(y, preds_int).fit()
model.summary()
```

Out[173]:

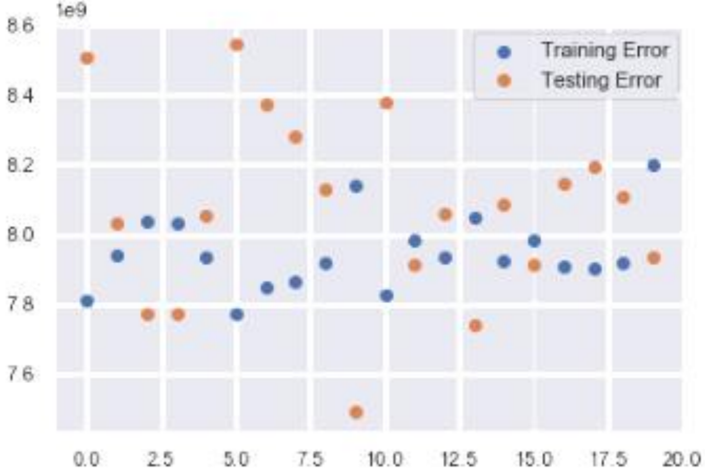
OLS Regression Results

Dep. Variable:	price	**R-squared:**	0.842
Model:	OLS	**Adj. R-squared:**	0.841
Method:	Least Squares	**F-statistic:**	1050.
Date:	Sun, 27 Jan 2019	**Prob (F-statistic):**	0.00
Time:	09:41:27	**Log-Likelihood:**	-2.6603e+05
No. Observations:	20756	**AIC:**	5.323e+05
Df Residuals:	20650	**BIC:**	5.331e+05
Df Model:	105		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	-2.782e+05	1.79e+04	-15.500	0.000	-3.13e+05	-2.43e+05
grade_(8, 9]	1.2e+05	2744.073	43.747	0.000	1.15e+05	1.25e+05
sqft_living	3.84e+05	1.09e+04	35.093	0.000	3.63e+05	4.05e+05
grade_(9, 10]	1.946e+05	4037.820	48.187	0.000	1.87e+05	2.02e+05

```
train_test_split(stepwise_X, y, test_size = 0.25)
```

MSE After Dropping Ineffective Variables



Price Predictor Model v1.0

In [122]:
```
###MSE is the squared price error
from math import sqrt
pos_MSE = -1*cv_20_results
Median_Error = sqrt(pos_MSE)
print(f'Average error of prediction model: ${round(Median_Error, 2)}')
```

```
Average error of prediction model: $89977.49
```

Highlights

Get the Square Footage

It is highly correlated with price

Zip Code Hotspots

Consider properties in the top 5 correlated zip codes to be more accurate predictions

Accuracy of Model

Plus or minus $90,000

Further Findings

Geo Map

Better visualization and more accurate geographical correlations

Adjusted for inflation

A huge variable not addressed by the current model

More Widespread

More samples from under represented neighborhoods

Thanks for your time