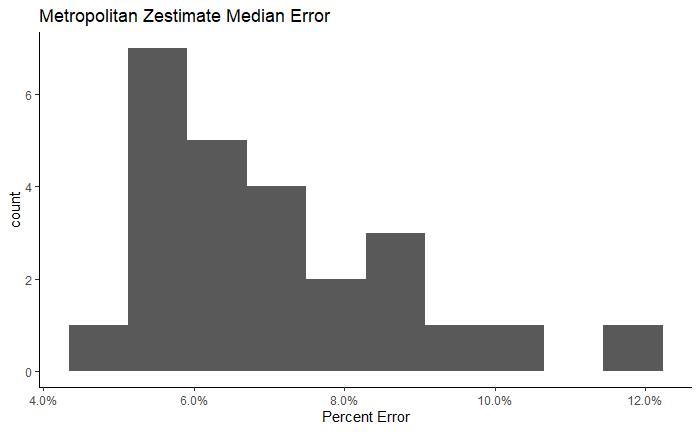
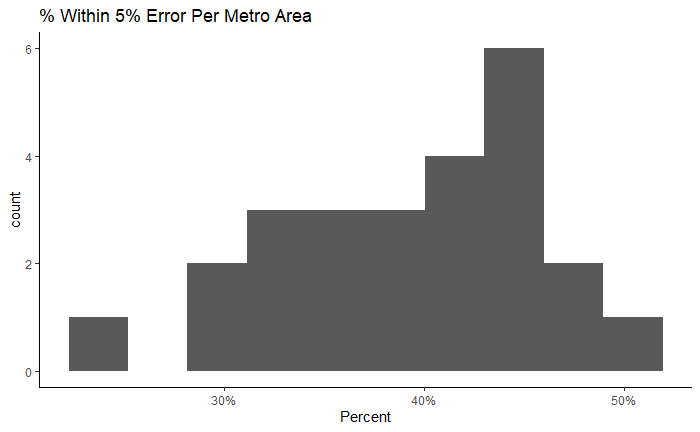
● Introduction/Motivation:

Housing and real estate has been on the horizon for data science and analytics for years. As mentioned in our proposal, Zillow, a major player in the housing space, has attempted to create machine learning models to predict housing prices and find arbitrage in prices. Zillow eventually scrapped the project saying large scale market prediction was nearly impossible. The company currently does have a value called ‘zestimate’ which predicts the assets value. These graphs below show Zillow’s public modeling error:



Zillow also has features such as ‘rent estimator’ which looks at previous values and predicts whether the rent is above or below market value. As it currently sits today, the housing market is skyrocketing to unseen heights.

Our group overall disagrees with Zillow and believes that through leveraging the power of Spark and other distributed computing we can account for fluctuations in the market. We are using Kaggle housing price data sampled from the Pacific Northwest of The United States. We believe that starting with a small model and gradually scaling into larger data.

● Problem: Formulate your problem as a data analytics or modeling problem.

Our problem is ultimately simple: predict asset prices based on asset attributes and location. We considered many loss metrics but ultimately settled on root mean squared error. Root mean squared error provides the most understand error as this the loss will be in context of dollars. Residuals, likewise, will be understandable.

Colab, scaling, cloud computing, single scripts

● Methods:

Data Cleaning:

The data needs some, but minimal cleaning. The columns seem easy to engineer, with plenty of opportunity for additional feature creation. Given the geographically localized data, our main objective will be to use a regression model to predict housing prices. This entails us minimizing the loss function of mean squared error. This process will be completed using PySpark which will allow us to handle vast amounts of data in a distributed environment. Additionally, modeling will be done using PySpark’s MLlib package.

Feature Engineering:

Modeling:

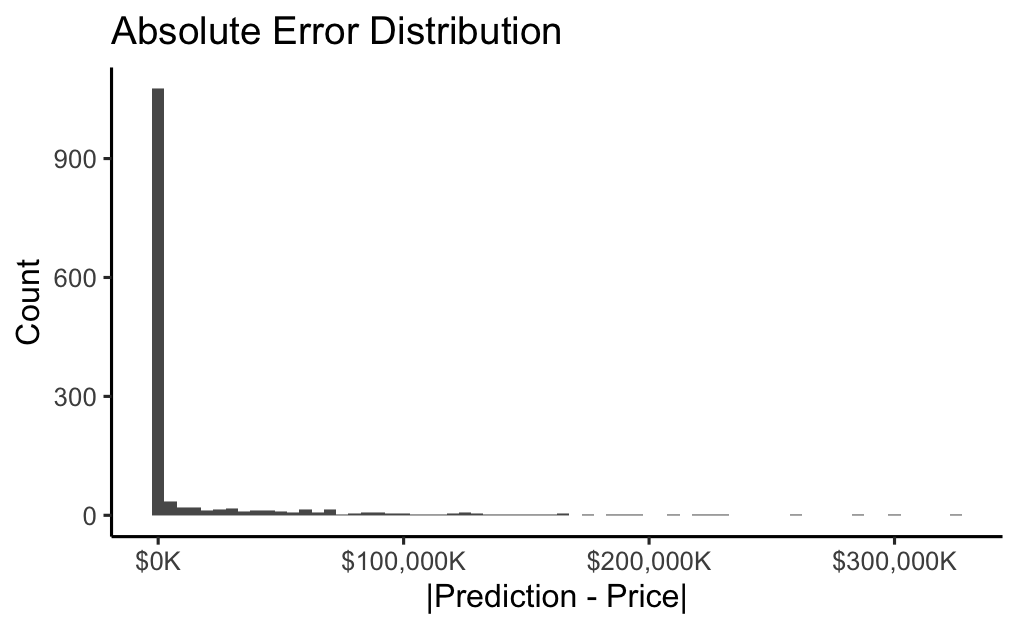
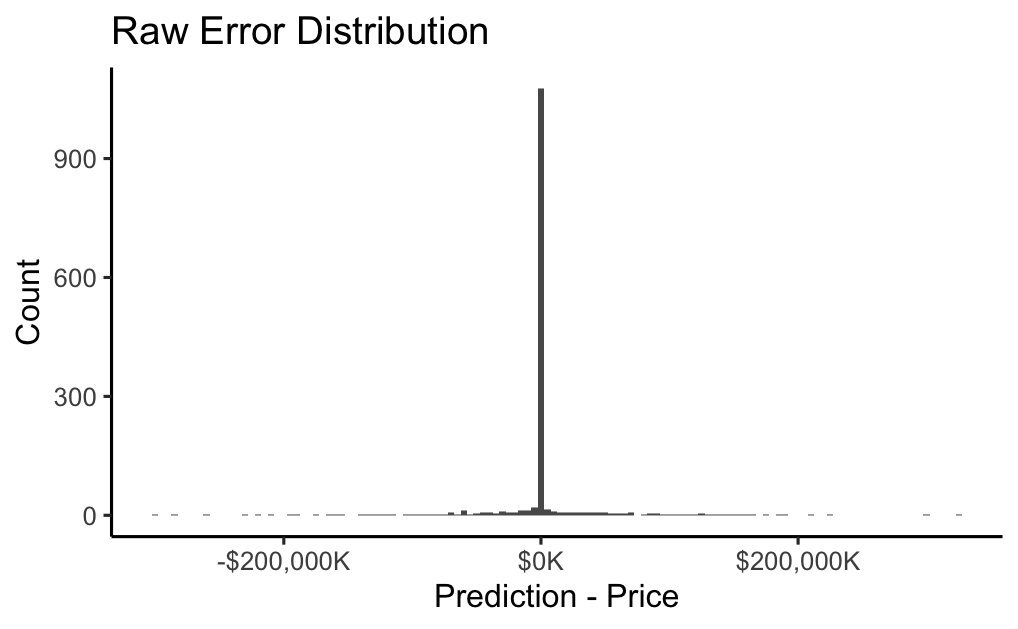
o Describe the algorithm or approaches you designed

o The tools and software used to implement the algorithm

o What problems did you face? How did you solve them?

● Results:

Our raw results were surprisingly very good with 90% of our predictions being within $5,000 of the actual price and 85.5% of our predictions being within 5% of the actual price. The average error in our predictions was $435.80 while the median was -$2.60. Since the median and mean differ by a large amount we assumed that there are some large error values that skew those results. This is represented by minimum error of -$301,100.70 and maximum error of $325,265.80. We also analyzed the absolute error to disregard the direction of the error in our predictions. The median absolute error was $234.40 and the mean absolute error was $13,963.70. Once again, these results are skewed due to the errors over $300,000. Below are histograms displaying both the raw error and absolute error. The bidwidths for both graphs is $5,000.



We further investigated our predictions to see if there were specific scenarios where our error spiked. Unfortunately there are no correlations between the characteristics of a house and the error amount. To extend our investigation even further, we filtered on absolute errors over a $100,000 threshold and there were still no significant relationships. In fact, we found that the percent error was negatively correlated with many of the house features when the absolute error amount was over $100,000.

