9/8/2023

Isom Brown

Western GOvernors University

Predicting Housing Prices via Applied Machine Learning

September 8, 2023

C964: Computer Science Capstone

Task 2 parts A, B, C, and D

Part A: Project Proposal for Business Executives 3

[Letter of Transmittal 3](#_Toc1044087116)

[Project Recommendation 4](#_Toc1721750458)

[Problem Summary 4](#_Toc205640471)

[Application Benefits 5](#_Toc739883765)

[Application Description 5](#_Toc1967641075)

[Data Description 5](#_Toc1580263866)

[Objectives and Hypothesis 6](#_Toc1084915425)

[Methodology 6](#_Toc1455093807)

[Funding Requirements 8](#_Toc1716359054)

[Data Precautions 8](#_Toc821607173)

[Developer’s Expertise 8](#_Toc742297278)

[Part B: Project Proposal 10](#_Toc1516064891)

[Problem Statement 10](#_Toc602847551)

[Customer Summary 10](#_Toc411678477)

[Existing System Analysis 10](#_Toc1668713332)

[Data 11](#_Toc1911399030)

[Project Methodology 12](#_Toc1307313575)

[Project Outcomes 13](#_Toc1967172832)

[Implementation Plan 14](#_Toc306674583)

[Evaluation Plan 16](#_Toc849271423)

[Resources and Costs 17](#_Toc1695791095)

[Timeline and Milestones 17](#_Toc787992355)

[Part C: Application 19](#_Toc1767968421)

[Part D: Post-implementation Report 20](#_Toc22579049)

[A Business (or Organization) Vision 20](#_Toc1496854501)

[Datasets 20](#_Toc645781746)

[Data Product Code 22](#_Toc157285634)

[Objective (or Hypothesis) Verification 25](#_Toc1315546441)

[Effective Visualization and Reporting 26](#_Toc656892821)

[Accuracy Analysis 30](#_Toc1956352798)

[Application Testing 32](#_Toc871280264)

[Application Files 33](#_Toc1639588139)

[User Guide 33](#_Toc1125023771)

[Summation of Learning Experience 35](#_Toc1702287338)

# Part A: Project Proposal for Business Executives

## Letter of Transmittal

Isom Brown, Sr. Software Engineer

253-282-2108 |ibrow71@wgu.edu | Seattle, Washington 98128

September 8, 2023

Vincent Prasad, Chief Operating Officer

123-456-7890 | vincent.prasad@email.com | Seattle, Washington 98128

Subject: Proposal for a new feature on the company website/app

Hi Vincent,

Over the past year or so, I and several other team members have noticed Noble Real Estates' growth stagnate. In looking at competing brokerages it's clear to me that while we offer a quality service, we lack a price estimation tool to show customers what a fair market price for a home would be. To combat this deficiency, I began developing “Noble’s Estimate”.

“Noble’s Estimate” will be similar to price estimators from top competitors like Redfin and Zillow. The feature will be developed with the use of machine learning models that can accurately and effectively predict housing prices across a wide range of real estate markets. By developing an efficient price estimator, I believe we will make Noble Real Estate the “go-to” online real estate brokerage and see new growth in the customer base.

During my undergraduate studies, I took several machine learning classes and developed a firm grasp of the subject matter. I think if we’re able to contract a couple of more full-time software engineers and a data scientist, we’d be able to quickly develop a working version of a housing price estimator. I know you are likely concerned about the cost of a project like this. However, I think you’d be happily surprised at the cost. After running the numbers, I was able to estimate the total cost of the project would be $116,600.

Thanks for taking the time to read over this letter and let me know what you think about this project.

Respectfully,

Isom Brown

## Project Recommendation

### Problem Summary

Noble Real Estate was founded in 2010 in the Pacific Northwest. Since its founding, it has worked to establish itself as a premier real estate brokerage. As the company has grown and expanded across the nation it’s needed a way to compete with the largest online real estate brokerages in the nation. In polling current and prospective customers, the primary thing separating Noble Real Estate and other national real estate brokers is a property estimate tool. This project aims to develop a machine learning program that can accurately and effectively predict housing prices across a wide range of real estate markets, to make Noble Real Estate the first choice for consumers when picking an online real estate broker.

### Application Benefits

Currently, when Noble Real Estate customers visit our website or application, they can only view the listed price for homes for sale on the market and the previous sale price of homes off the market. For customers to get a price estimate for the home they’re interested in, they are forced to turn to our competitors or pay to have a formal appraisal done on the home. This causes Noble Real Estate to miss out on potential customers who would’ve purchased homes through the brokerage or used one of its auxiliary services. By implementing a price estimator tool Noble Real Estate would likely retain these customers. It also will act as a draw for new customers who have never considered using Noble Real Estate before.

### Application Description

Noble Real Estate’s price estimator will be based on a machine learning model that can predict or estimate housing prices based on the features of the home. The model will accomplish this by training on public data sets of past home sales. As the model finishes training and is finely tuned it should be able to provide accurate price estimations for customers for homes currently listed on the market. It will even be able to allow users to enter the features of a home not listed on the market and provide a price estimation for that property. Solving the customers' need for an alternative price estimation.

### Data Description

The data for this project will be sourced from publicly available sources. For initial training and testing of the model, I’ve chosen to use a data set from Kaggle.com looking at house prices from Seattle, WA. All the features listed in the data set are quantitative and include features like the square footage of a home, the number of bedrooms, and the zip code for the address of the home. The data is structured as a comma-separated value file, something similar to that of an Excel spreadsheet. To look at how the features of the home impact its price I’ll be using several graphs in “Part C: Application” which show the interdependencies of the data. The data will be cleaned and processed before training the machine learning model to ensure any anomalies are removed.

### Objectives and Hypothesis

Noble Real Estates' objective is to be to estimate real estate prices with an error rate below 2.09% for homes on the market and an error rate below 6.34% for homes off the market. Creating a real estate price prediction model this accurate would allow Noble Real Estate to compete with Redfin and Zillow. To test this hypothesis, we will gather a large amount of data and split it up to allow for proper training and testing of our model. The model will be evaluated with “real world” data to test our hypothesis.

### Methodology

For this proposed project I decided to go with the Cross-Industry Standard Process for Data Mining or CRISP-DM. The primary reasoning for this was the fact CRISP places a larger emphasis on data mining and on clear phases and objectives, similar to that of the waterfall method. Both of which align with Noble Real Estates' goals. An application of CRISP-DM to this project can be seen below.

* **Business Understanding**: As previously mentioned in “Objectives and Hypothesis” the goal of this project is to create a user-friendly application that can accurately estimate the value of homes.
* **Data Understanding:** The data for this project will be sourced from publicly available sources. For initial training and testing of the model, I’ve chosen to use a data set from Kaggle.com looking at house prices from Seattle, WA. All the features listed in the data set are quantitative and include things like the square footage of a home, the number of bedrooms, and the zip code for the address of the home. The data is structured as a comma-separated value file, something similar to that of an Excel spreadsheet.
* **Data Preparation:** Data must be cleaned and transformed so it can be placed in a data frame. Any outliers and anomalies must be removed to ensure the machine learning model is properly trained.
* **Modeling:** Based on the research conducted during the lead-up to this project, I determined a regression model would be the best option for this real estate price estimator. During the modeling phase, it's important that we properly tune the model to ensure we achieve our accuracy goals without overfitting the model to our training data.
* **Evaluation:** During the evaluation phase new testing data is of utmost importance to determine if our model is making accurate predictions. Thankfully besides being able to just set aside testing data, there is a constant influx of new homes on the market to provide new training/testing data if needed.
* **Deployment:** Once the model has been properly tuned, it must be integrated into Noble Real Estate's current website and mobile application. After this is completed, it can be deployed to the end-user and monitored for maintenance and future improvement opportunities.

### Funding Requirements

When it comes to the funding requirements of this project the primary cost will be labor and this will vary depending on the time it takes to complete the project. Based on the scope of this project Noble Real Estate would likely need to contract two full-time software engineers with machine learning experience and a full-time data scientist, for 3 months to complete the project. The average yearly salary for a software engineer is $120,000 and the average yearly salary for a data scientist is roughly the same. This means over the 3 months the labor costs would roughly be $90,000. Infrastructure costs will include computers, cloud services/licenses, and the development of a data pipeline. Based on past projects I estimated the cost for these items would roughly be $11,000. Once the project is completed, monitoring and maintenance of the program must be done. This could be handled internally for roughly $5,000 a year. Lastly, money must be set aside in the budget for any contingencies or unforeseen issues. The contingency budget will be 10% of the project, which comes out to $10,600. This brings the total cost of the project (including the contingency reserve) to $116,600.

### Data Precautions

The data for this project will come from a public data set. As a result, none of the data is sensitive or protected and no precautions need to be taken.

### Developer’s Expertise

Why am I qualified to lead this project? Since earning my bachelor’s degree in computer science I spent the last few years working as a software engineer at some of the top online real estate brokers in the nation. During that time I developed an interest in artificial intelligence and machine learning. I took this passion and followed it, teaching myself the fundamentals of machine learning through online courses and academic reading. Knowing I’d like to apply this passion to my real estate work I focused on studying regression models and how other real estate brokers had developed their machine learning programs for price estimation. I think this combined with my inside knowledge of the online real estate brokerage business, and my willingness to learn and find alternative solutions to problems, makes me the ideal engineer to bring this concept to fruition.

# Part B: Project Proposal

## Problem Statement

Based on customer surveys and industry research I determined Noble Real Estates' lack of growth was tied to its inability to provide price estimations for properties. This is something our main competitors like Redfin and Zillow, have been doing with the help of machine learning for several years now. Both companies have been developing their models for so long that they’re able to predict the price of homes on the market within an error rate below 2.09% and an error rate below 6.34% for homes off the market. To compete with online brokers Noble Real Estate needs to develop its machine learning models based on supervised learning. The models will have to analyze data using some sort of regression model to accurately estimate property prices.

## Customer Summary

The clients or customers of Noble Real Estate include a diverse range of individuals, each with their own unique needs and preferences. Someone interested in using a real estate price estimation tool would be anyone interested in real estate, whether that’s buyers, sellers, or you typical HGTV fan interested in browsing real estate. As mentioned previously recent surveys were conducted on individuals leaving Noble Real Estates website and they were asked for feedback regarding what the company could do better. The number one piece of feedback that individuals left was that they’d like to see Noble Real Estate provide a price estimation tool. By including this estimation tool Noble Real Estate hopes to increase the number of customers using its services.

## Existing System Analysis

Noble Real Estate’s current system operates through online platforms and technology rather than traditional brick-and-mortar offices. It offers customers several services including access to property listings, the ability to view photos and property details, schedule viewings, and conduct various real estate transactions. It also links customers to real estate agents and mortgage lenders. However, it lacks any real data analytics. This is where our new real estate price estimator comes into play. With the help of machine learning the price estimator will provide clients with insights into property valuations, market trends, and investment opportunities.

## Data

The data used for this project will be collected from Kaggle.com. The website was developed by Google as a data science competition platform. It posts publicly available data sets for research and learning purposes. The data set I chose is easily accessible through Kaggle and downloaded as a CSV file. This CSV file can then be read by our program into a data frame which can then be manipulated and transformed.

One of the reasons this data set was chosen was that it is very complete and there were few outliers. However, there were still some inconsistencies with the data. The main issue I observed was that several homes listed in the data set were missing values for the “lot size” feature. In going through this data, I assumed that these properties were most likely apartments, therefore their lot size would be zero. As a result, I converted all missing values in the “lot size” feature column to zero.

Another issue with the data was that the “lot size” feature was listed in acres and square feet. This meant there was another feature column needed to label the unit of measurement for the lot size. To simplify the data set all “lot sizes” were converted to square feet and the feature column labeling the unit measurement for lot size was dropped.

## Project Methodology

For this proposed project I decided to go with the Cross-Industry Standard Process for Data Mining or CRISP-DM. The primary reasoning for this was the fact CRISP places a larger emphasis on data mining and clear phases and objectives, similar to the waterfall method. Both of which align with Noble Real Estates' goals. An application of CRISP-DM to this project can be seen below.

* **Business Understanding**: As previously mentioned in “Objectives and Hypothesis” the goal of this project is to create a user-friendly application that can accurately estimate the value of homes.
* **Data Understanding:** The data for this project will be sourced from publicly available sources. For initial training and testing of the model, I’ve chosen to use a data set from Kaggle.com looking at house prices from Seattle, WA. All the features listed in the data set are quantitative and include things like the square footage of a home, the number of bedrooms, and the zip code for the address of the home. The data is structured as a comma-separated value file, something similar to that of an Excel spreadsheet.
* **Data Preparation:** Data must be cleaned and transformed so it can be placed in a data frame. Any outliers and anomalies must be removed to ensure the machine learning model is properly trained.
* **Modeling:** Based on the research conducted during the lead-up to this project, I determined a regression model would be the best option for this real estate price estimator. During the modeling phase, it's important that we properly tune the model to ensure we achieve our accuracy goals without overfitting the model to our training data.
* **Evaluation:** During the evaluation phase new testing data is of utmost importance to determine if our model is making accurate predictions. Thankfully besides being able to just set aside testing data, there is a constant influx of new homes on the market to provide new training/testing data if needed.
* **Deployment:** Once the model has been properly tuned, it must be integrated into Noble Real Estate's current website and mobile application. After this is completed, it can be deployed to the end-user and monitored for maintenance and future improvement opportunities.

## Project Outcomes

The goal of this project is to develop a standalone price estimation tool that can be incorporated into Noble Real Estates website and app in the near future. The tool will use a machine learning model to estimate the price of a home given the following features of the home: number of bedrooms, number of bathrooms, the square footage of the home, the lot size of the property, and the zip code the property is located in. The tool will include histograms showing distributions of the data features and a scatter matrix demonstrating data correlations. It will allow users to get an estimate for the price of a home by inputting the previously mentioned features and let users know how accurate its estimations are.

**User Guide:**

1. Open a web browser (Chrome, Microsoft Edge, etc.)
2. Go to <https://docs.conda.io/projects/miniconda/en/latest/> and download the latest version of Miniconda for your operating system
3. Download provided c964\_capstone.zip from task submissions
4. Extract the zip to the computer note the folder name and path
   1. Ex.) “C:\Users\Your\_Username\Desktop\c964\_capstone”
5. Open the recently downloaded Miniconda terminal. The actual name for the app will be “Anaconda Prompt (anaconda3)”
6. Change your current directory in the Miniconda terminal to “c964\_capstone” by entering
   1. Ex.) “cd C:\Users\Your\_Username\Desktop\c964\_capstone”
7. Enter “conda create --prefix ./env pandas numpy matplotlib scikit-learn” to have Miniconda install necessary libraries and packages
8. Type “y” to and hit enter to proceed
9. Activate your conda environment by entering the following into the command line
   1. Ex.) “conda activate C:\Users\Your\_Username\Desktop\c964\_capstone\env”
10. Install Jupyter Notebook to your environment by entering the following into the commandline
    1. “conda install jupyter”
11. Next open Jupyter Notebook by entering the following into the command line
    1. “jupyter notebook”
12. A window should open in your browser for Jupyter Notebook. Click on the “C964PartC.ipynb” file.
13. Once “C964PartC.ipynb” is open click the double play arrow to restart the kernel and run all cells
14. Look through the application and follow any prompts

## Implementation Plan

As mentioned in the “Project Methodology” Part B, the general strategy for the development of the project will be a combination of an agile framework and CRISP-DM. This will be implemented in the project by creating a generalized rollout plan for the project, breaking the project into different phases or sprints that last a week to a few weeks at a time. Since we’re trying to follow the CRISP-DM method that means the project would start with gaining a business understanding of the project. This would include developing objectives for the project and a hypothesis for the outcome. We’ve already done this for the most part with the development of this documentation.

The next sprint or phase would be to develop an understanding of the data. This would include determining how we were sourcing our data. Once we have our data, we need to be able to describe and understand the features listed in it, how it is formatted, the number of records, etc. After this we can explore the data and begin to analyze it more, looking at relationships and correlations between features and the target we are estimating.

Once an understanding of the data has been obtained, the next phase of the project would be to clean and prepare the data. Typically, this would include removing any outliers within the data, and removing and or adding features to the data set.

With all the data sourced, prepped, and cleaned the modeling of the project can begin. Based on the research conducted during the lead-up to this project, I determined a regression model would be the best option for this real estate price estimator. During the modeling phase, it's important that we properly tune the model to ensure we achieve our accuracy goals without overfitting the model to our training data.

After our model is completed, an evaluation of the overall project and model can begin. This would include getting new housing data to test our model to ensure it was not overfitted to our training data. It would also include unit testing. The last phase or sprint of the project would be to integrate the price estimating tool into Noble Real Estate's current website and mobile application. After this is completed, integration and system testing could be completed. The project could then move to user acceptance testing before finally being deployed. Once deployed it will be monitored for maintenance and future improvement opportunities.

As far as dependencies go for the project, most of the project follows a simple finish-to-start dependency. This means that one task must finish before the next can start. As far as external dependencies our only concern would be for our data. The data set we have chosen for the project has a limited number of homes to train our model on. To make the machine learning model and overall application more robust in the future a data pipeline could be created with Multiple Listing Services (a real estate listing website) API to have a constant influx of new housing data.

## Evaluation Plan

Throughout the project, different verification methods will be used at each stage of development to ensure the project is on track to meet its goals. During the business understanding phase and initial project planning, a project proposal will be developed. This plan will be reviewed by stakeholders and needs to receive stakeholder approval before the project can move forward. After this has happened, we can move on to the next phase of the project, and develop an understanding of the data. The evaluation that would occur during this section would include an inspection of the data frame design and possibly a peer review of the exploration of data. The next phase of the project would be the cleaning and preparation of data. To evaluate this section of the project code reviews will be conducted. Once this has been completed, we’re at the modeling phase of the project. Evaluations used during this section include statistical analysis of the model’s outcome including mean absolute error (MAE) and the coefficient of determination. To go along with the statistical analysis, unit testing, and code reviews would be conducted. The next phase of the project would be evaluation. After the model is completed an evaluation of the overall project and model can begin. This would include the last phase or sprint of the project, which would be to integrate the price estimating tool into Noble Real Estate's current website and mobile application. At this point integration and system testing could be completed. The project could then move to user acceptance testing before finally being deployed.

## Resources and Costs

**Estimated hardware and software costs**:

* Development machines - $7,500
* Server/cloud resources - $2,500
* Windows operating system fees - $600
* Jupyter Notebook – free
* Additional IDEs and text editors - $150

**Estimated labor and time costs**:

* Two full-time software engineers (3 months of pay) - $60,000
* One full-time data scientist (3 months of pay) - $30,000
* Ongoing support and maintenance - $5,000
* Contingency reserve - $10,600

## Timeline and Milestones

|  |  |  |
| --- | --- | --- |
| **Milestone** | **Start** | **End** |
| Project initiation | 8/28/2023 | 9/1/2023 |
| Data Collection and Analysis | 9/4/2023 | 9/15/2023 |
| Data Splitting and Processing | 9/18/2023 | 9/29/2023 |
| Model Training | 10/2/2023 | 10/6/2023 |
| Evaluation | 10/9/2023 | 10/27/2023 |
| Deployment | 10/30/2023 | 11/5/2023 |

# Part C: Application

See submitted “c964\_capstone” zip file and User Guide listed in Part D. For a breakdown of the files contained in the zip file turn to the Application Files in Part D.

.

# Part D: Post-implementation Report

## A Business (or Organization) Vision

The problem faced by Noble Real Estate was a continued lack of growth over the last year. Based upon customer surveys and industry research we determined customers were turning to competitors, including Redfin and Zillow, for the price estimation tools they provided. Noble Real Estate determined the simple solution to the problem would be to develop its own price estimation tool through the use of machine learning. Customers of Noble Real Estate can now use the recently developed price estimation tool to see how features of a home can impact a home's price. It also allows users to enter the features of a home off the market and be given a price estimate for the property. For example, if a customer knows the number of bedrooms, number of bathrooms, square footage of the property, lot size of the property, and the zip code of the property; they’ll simply be able to enter this information into the tool and be given a price estimate for the property. During beta testing of the price estimation tool, customers advised it would sway them to use Noble Real Estate over other brokerages. This helps to prove the price estimation tool likely will be able to increase Noble Real Estates' growth once released.

## Datasets

The data used for this project was obtained from Kaggle.com. The website was developed by Google as a data science competition platform. It posts publicly available data sets for research and learning purposes. The data set is easily accessible through Kaggle and downloaded as a CSV file. This CSV file can be placed in the folder where our program is being stored, and then be read by our program into a data frame which can then be manipulated to transform and clean up the data set. An example of what the raw CSV file looked like can be seen below.

A screenshot of a computer

Description automatically generated

One of the reasons this data set was chosen was that it is very complete and there were few outliers. However, there were still some inconsistencies with the data. The main issue I observed was that several homes listed in the data set were missing values for the “lot size” feature. In going through this data, I assumed that these properties were most likely apartments therefore their lot size would be zero. As a result, I converted all missing values in “lot size” to zero.

Another issue with the data was that the “lot size” feature was listed in acres and square feet. This meant there was another feature column needed to label the unit of measurement for the lot size. To simplify the data set all “lot sizes” were converted to square feet and the feature column labeling the unit measurement for lot size was dropped. Another column that was dropped was the feature column “size units”. All this feature did was label the unit of measurement for the square footage of the actual home which was deemed unnecessary. Below is an example of what the processed data looked like inside a data frame.

A screenshot of a computer

Description automatically generated

The raw CSV file can be accessed through the provided compressed zip file named “c964\_capstone”.

## Data Product Code

As mentioned in the previous section one of the primary reasons the housing data set from Kaggle was chosen was that the set required little processing. However, some changes were made. Once the data set had been moved into a data frame, I looked for any elements within the data frame that were missing data (“na”). Only a few hundred of the homes listed in the data set were missing values and for these homes, it appeared they were only missing lot size measurements and a unit of measurement for the missing value.

In reviewing the data further, I assumed these properties were likely apartments or condos and that’s why their lot sizes were missing. To deal with the issue all properties with “na” for a lot size were set to zero and their unit of measurement was changed. This was simply done through the fillna() function built into the panda's library. See the code snippet below.

A screen shot of a computer code

Description automatically generated

One of the last big changes to data was converting all of the lot sizes to the same unit of measurement. In the original data set lot sizes were listed in acres and square feet. This required there to be another feature column listing the unit measurement for the lot size. This appeared unnecessary and I knew the data frame would be better suited for modeling if all the elements/features were numeric. To solve this problem, I simply converted any lot size with acreage as a unit of measurement to square feet.

With this done the feature column listing the unit of measurement for the lot could be dropped. I also dropped the feature column listing the unit of measurement for the square footage of the home because all values were in square feet already.

Once the data had been cleaned and transformed, I began using descriptive methods to analyze it. The descriptive methods used included histograms and a scatter plot matrix. Using the descriptive methods, I observed some trends within the data. The first is that the square footage of the home and the price of the home seemed to be highly correlated. An increase in the size of the home tends to lead to an increase in price. See below scatter plot.

A blue and red dot graph

Description automatically generated with medium confidence

I also observed that the price of homes varies throughout the different zip codes I had in my data set. See the histogram below.

A graph with numbers and a line

Description automatically generated with medium confidence

When choosing a non-descriptive method, I knew I needed to use a supervised machine learning model because I had labeled inputs and outputs. I also knew I needed to choose a machine learning model using regression given I was trying to predict a value. While researching different types of regression models, I read several scholarly articles about machine learning techniques for housing price predictions. Some of the articles reviewed included: “*House Price Prediction using Random Forest Machine Learning Technique”*, “*Predicting Owner-Occupied Housing Values Using Machine Learning: An Empirical Investigation of California Census Tracts Data”*, and “*Housing Price Prediction via Improved Machine Learning Techniques”*. Each of these articles provided me with a unique insight into designing a real estate price estimator using machine learning. However, each of the articles came out with a different outcome on what regression model should be used when trying to predict housing prices. The regression models listed included Random Forest, Lasso, Elastic Net, and Stack Generalized Regression.

Ultimately, I decided to try several machine learning models to see which worked best. To begin testing the models I separated my data into a feature matrix and a target matrix (variables x and y). I then split each matrix further into training and testing data, with 80% of the data being used for training and the other 20% being used for testing. See below code snippet below.

A screenshot of a computer code

Description automatically generated

With the data frame split up into training and testing data, I created several different machine learning models to see which could predict housing prices the best. I ended up testing several different regression models including Random Forest, Linear, SGD, LASSO, Elastic Net, GradientBoosting, and Ridge regression. All models were scored using the coefficient of determination (R²). This is a number between 0 and 1 that measures how well a statistical model predicts an outcome. Much to my surprise Linear Regression scored the best when I compared the models, even when models like Random Forest Regression were tuned. As I result, I decided to use a Linear Regression model for the project.

## Objective (or Hypothesis) Verification

Noble Real Estates' objective was to estimate real estate prices with an error rate below 2.09% for homes on the market and an error rate below 6.34% for homes off the market. Creating a real estate price prediction model this accurate would allow Noble Real Estate to compete with Redfin and Zillow. To test this hypothesis, we gathered a large amount of data and split it up to allow for proper training and testing of our model.

The objective of this project was not met. When looking at the coefficient determination of the Linear Regression model I created, I observed it typically produced a score of 0.58. This indicates the model was only able to gain a moderate understanding of the data. A coefficient determination score seemed vague when thinking about how accurate a prediction model can be. To help with this I also scored the model by looking at the mean absolute error (MAE). When looking at the MAE I observed that on average the model was predicting home values $200,000. This was nowhere close to the accuracy Redfin or Zillow were predicting home prices.

When researching possible reasons why my machine learning model had failed to meet the objective I set for it I discovered a couple of things. The first was that Redfin and Zillow use over 500 data points for a single home to be able to predict the price of the home. This is 100 times as many features as each home had in my data set. Another thing I learned was that I was working with a limited amount of data to train my model. For training the model I had roughly 1,600 homes. When I compared this to the “California Housing” data set (a data set typically used for individuals studying housing predictions on Kaggle) I observed this data set had close to 20,000 homes for training and testing models.

## Effective Visualization and Reporting

In the development of this project data exploration and analysis were conducted. The primary goal had been to discover any missing values, outliers, and or anomalies within the data. The secondary goal had been to look for trends with the data to see how the features of a home influenced the price of a home. During my data exploration and analysis, I used several charts and graphs to look at the data, as well as a simple function to check for missing values within my data. The charts and graph primarily included histograms and scatter plots.

The function used was an “isna()’, and it is a built-in function in pandas. The function detects missing values within a data frame. When used in combination with the “sum()” function, “isna().sum()”, I’m able to print to the console a simple list of the column names and the number of missing values within the column. See the code snippet below.A screenshot of a computer

Description automatically generated

After I was able to take care of the missing values discovered with the code above, I turned to my graphs and charts to begin exploring and analyzing the data. During this process, I used a combination of histograms and scatter plots. Below is a picture of the first scatter plot used.

A blue and red dot graph

Description automatically generated with medium confidence

My common sense led me to think there had to be some sort of correlation between the size of the home and its actual price. The scatter plot pictured above was created to examine the relationship between the square footage of the home and its price. It was clear to me from the scatter plot that the bigger the interior of the home the higher the price of the home is likely to be. Of course, there are exceptions to this rule as can be seen within the data.

The next thing I looked at was a series of histograms. The histograms showed how different features of homes were spread across the data set. For instance, how many of the homes in the data set had two bathrooms, or how many of the homes in the data set were located in a particular zip code. A picture of the histograms can be seen below.

A graph of different sizes and colors

Description automatically generated with medium confidence

While examining the histograms one of the things that stood out most to me was that most of the houses in my data set appeared to come from two zip codes.

After looking at the series of histograms I looked at a scatter matrix of the data. The purpose of this was to look at relationships between features of the home. When looking at the scatter matrix there were some obvious correlations between features. For instance, the more square footage a home has the more bedrooms or bathrooms it will have. Something interesting I observed while looking at the scatter matrix was that the size of homes varied in the provided zip codes. See a picture of the scatter matrix below.

A graph of blue and white lines

Description automatically generated with medium confidence

## Accuracy Analysis

To assess the accuracy of my machine learning model I looked at its coefficient determination score () and looking at its mean absolute error (MAE) score. The coefficient determination is a score that shows how well a model understands the feature's impact on the target feature or value you’d like to predict. During testing, the Linear Regression model I chose to use had a coefficient determination score of 0.58 which shows that the model has learned a good amount about how the independent variables going into the model affect the outcome.

After looking at the coefficient determination score, I looked at the mean absolute error. This looks at the absolute value of the margin of error between the target value and the model's prediction. During testing the model had a MAE of roughly 200,000. This meant on average the home prices estimated by the model were off by $200,000. To gain a visual understanding of what this meant I graphed the model’s prediction of home prices vs. their actual prices (see the below image).

A graph of red dots

Description automatically generated

## Application Testing

Before testing the application, I made sure I had a clear understanding of the project requirements. I then researched different testing metrics to determine how to measure my machine learning model's accuracy, so I could clearly define my own goals for the project. After this, the code was developed. The code was tested one code segment at a time, like a miniature unit test. Once I had gone through each code segment, I combined everything and tested the entire application to make sure it was functional. When I knew the application was functional, I developed small test cases to make sure it would meet the “end users” needs. The main improvements came about while training and testing my machine learning model. This was primarily done through a guess and check system, comparing the accuracy scores of each model. This testing ultimately helped me decide which machine learning model to use for my application.

## Application Files

\c964\_capstone

\ .ipynb\_checkpoints

\housing\_prices

\train.csv

\C964PartC.ipynb

The project will be submitted as a compressed zip file named “c964\_capstone”, which contains necessary files for the project. Once the extraction of the zip file has occurred the structure of the folder will be the following. The root folder named “c964\_capstone” contains several folders and files including the “.ipynb\_checkpoints” which contains manually saved IPython Notebook files for version control, the “housing\_prices” folder which contains the “train.csv” file which is the comma separated value file containing my housing data from Kaggle.com, and the last file in the folder is the “C964PartC.ipynb” which is the IPython Notebook file which contains my source code for the program. For instructions on how to run the program please turn to the user guide.

## User Guide

1. Open a web browser (Chrome, Microsoft Edge, etc.)
2. Go to <https://docs.conda.io/projects/miniconda/en/latest/> and download the latest version of Miniconda for your operating system
3. Download provided c964\_capstone.zip from task submissions
4. Extract the zip to the computer note the folder name and path
   1. Ex.) “C:\Users\Your\_Username\Desktop\c964\_capstone”
5. Open the recently downloaded Miniconda terminal. The actual name for the app will be “Anaconda Prompt (anaconda3)”
6. Change your current directory in the Miniconda terminal to “c964\_capstone” by entering
   1. Ex.) “cd C:\Users\Your\_Username\Desktop\c964\_capstone”
7. Enter “conda create --prefix ./env pandas numpy matplotlib scikit-learn” to have Miniconda install necessary libraries and packages
8. Type “y” to and hit enter to proceed
9. Activate your conda environment by entering the following into the command line
   1. Ex.) “conda activate C:\Users\Your\_Username\Desktop\c964\_capstone\env”
10. Install Jupyter Notebook to your environment by entering the following into the commandline
    1. “conda install jupyter”
11. Next open Jupyter Notebook by entering the following into the command line
    1. “jupyter notebook”
12. A window should open in your browser for Jupyter Notebook. Click on the “C964PartC.ipynb” file.
13. Once “C964PartC.ipynb” is open click the double play arrow to restart the kernel and run all cells
14. Look through the application and follow any prompts

## Summation of Learning Experience

During my time at WGU studying computer science, several classes helped me prepare for this project including Business of IT- Project Management, Advanced Data Management, and Intro to Artificial Intelligence. Business of IT – Project Management was extremely helpful when it came to the write-up portion of this project. Advanced Data Management gave me a basic understanding of things I should be looking for in a simple data set and how to clean/process data. The most helpful class was Intro to Artificial Intelligence because during this class I learned what machine learning was and was able to research different types of machine learning models I might use for this project.

In addition to WGU resources my WGU mentor recommended I go through Udemy’s “Complete Machine Learning & Data Science Bootcamp”. The course showed how to set up an environment in which a machine learning project could be implemented, commonly used Python libraries when developing your machine learning projects, and common problems you may run into when developing your project. Overall, this outside resource was extremely helpful and gave me a practical understanding of what a machine learning project could be like.

When I started this project, it felt very overwhelming being so open-ended. However, as I started focusing on the requirements and looking at the course resources provided by Dr. Jim Ashe of WGU, it started to seem somewhat achievable but still a long shot. It took me starting the project and working on the first few problems before I started realizing I could accomplish this goal. I think in life I deal with “paralysis by analysis” a lot and I think this project helped to show me that sometimes starting on a project or goal is the hardest part. Going forward in life I’ll do my best to remember this whenever learning something new or starting towards a new goal.

**References**

Adetunji, A. B., Akande, O. N., Ajala, F. A., Oyewo, O., Akande, Y. F., & Oluwadara, G. (2022). House Price Prediction using Random Forest Machine Learning Technique. *Procedia Computer Science*, *199*, 806–813. [https://doi.org/10.1016/j.procs.2022.01.100](https://protect-us.mimecast.com/s/RLcFCG6Xp5UWKBX1yi7b8Jo?domain=doi.org)

Bourke, D., & Neagoie, A. (2023, September 1). *Complete Machine Learning & Data science bootcamp 2023 | udemy*. Udemy. https://www.udemy.com/course/complete-machine-learning-and-data-science-zero-to-mastery/

Dutta, A. (2023, June 5). *Random Forest regression in Python*. GeeksforGeeks. https://www.geeksforgeeks.org/random-forest-regression-in-python/

*How to plot predicted values vs the true value*. Stack Overflow. (1966, February 1). https://stackoverflow.com/questions/58410187/how-to-plot-predicted-values-vs-the-true-value

Incze, R. (2019, September 14). *The cost of machine learning projects*. Medium. https://medium.com/cognifeed/the-cost-of-machine-learning-projects-7ca3aea03a5c

Redfin. (n.d.). *Redfin estimate*. About the Redfin Estimate | Home Value Estimator. https://www.redfin.com/redfin-estimate

E. Simlai, P. (2021). Predicting owner-occupied housing values using machine learning: an empirical investigation of California census tracts data. *Journal of Property Research*, *38*(4), 305–336. [https://doi.org/10.1080/09599916.2021.1890187](https://protect-us.mimecast.com/s/sI3PC82zR6tODX444u1pmqL?domain=doi.org)

Truong, Q., Nguyen, M., Dang, H., & Mei, B. (2020). Housing Price Prediction via Improved Machine Learning Techniques. *Procedia Computer Science*, *174*, 433–442. [https://doi.org/10.1016/j.procs.2020.06.111](https://protect-us.mimecast.com/s/RM0aC5ywO9cg5M727COiiJ-?domain=doi.org)

Redfin. (n.d.). *Redfin estimate*. About the Redfin Estimate | Home Value Estimator. https://www.redfin.com/redfin-estimate