vehOctober 14, 2023

C964: Computer Science Capstone Template

Task 2 parts A, B, C and D

**Part A: Letter of Transmittal**

Letter Template

Isaiah Ragland

10/4/2024

Machine Learning Proposal

Louisville, Ky

---

Tony Smith

Homes for Sale Inc.

135 Somewhere Dr

Dear Tony Smith,

It has come to our attention that the housing market has boomed in the past recent years. As of right now, there are minimal resources for a customer to gauge the price of the home they are looking for. Customers can inquire about the price of a specific home, but it may take a while for the real estate agent to get back to them with an answer. What if we can create an application using the data you have on file about homes in the area to predict the price a potential customer might be interested in instead of waiting for a callback, they can get their answers immediately. This application will use machine learning to use distinctive features of a home, such as bedrooms, bathrooms, square footage, etc. All the important things a potential customer will look at before making a purchase.

This application will benefit an organization by cutting down the number of inquiries customers are making, which will in turn cut down the time real estate agents spend on the phone

researching answers. This would give agents the extra time to complete additional work and take excess workload off of them since customers can directly access the application and get an estimate of prices they may be looking at when looking at a house. This could also save money on the current resources they may be using to find out this information.

We would like to create this application to help customers guide themselves on the right direction in purchasing a home. By giving them live results of the current market, this data will be directly connected to your database so that any updates to the data will reflect in the price prediction. By using machine learning we will be able to continuously update and retrieve information that will be valuable to sales and or to the seller. As far as ethical concerns, there will not be any since this is all data readily available to the public already.

Our estimated cost for this project will be estimated around $39,000 as a starting investment. The project itself should take no longer than 2-3 months to complete. Our data will drive straight from the data King County, Washington State area will provide. In my 8 years in the computer science/ machine learning field also combined it with the relevant real estate expertise I have in the field. We will be able to deliver an application that could change the way homes are bought and sold. We would like to be the first to do it for you.

We are looking forward to your feedback and opinion on the matter. Thank you for you time.

Sincerely,

Isaiah Ragland, Machine Learning Engineer/ Real Estate consultor

**Part B: Project Proposal Plan**

Project Summary

The current problem we are facing in the housing market is a rise in the housing market. The rise in inquiries that customers are making is making it difficult for agents to keep up with this and there additional tasks. We will be able to make a machine learning model to predict the price of what a home can be bought or sold for. Customers will have direct access to this resource as it directly pertains to them. By making this application, we will be able to help the customer without directly helping them, this will give our agents some time to do other work instead of responding to inquiries throughout work hours. Which will also in turn lead to a much better and efficient work environment as a whole. Our deliverables will be the application that we developed and a user interface the customer can use, such as selecting the number of bedrooms, bathrooms, the square footage they are looking for, and the location they might be interested in moving to. This will also include a UI and a user guide to show how to use this particular application.

As a real estate agent, there are a lot of things to deal with on a day-by-day basis of the job itself. When a customer inquiries about a particular property, the agents have to manually look it up in the database and make the best guess price on how much this home will either sell for or can be bought at. This can consume a lot of time throughout the day, especially since the booming of the housing market, a lot of potential customers want to buy houses for investments or for their growing families. With the development of this application, the agents will be lifted from some responsibility of having to respond to inquiries and have more time to do what they need to for their particular role, whether that be responding to emails or having additional meetings to discuss the next steps in the organization itself. This will also be beneficial to the customer since it is a self-serve application, and they are able to use it whenever they please. If they were to have any additional information about a property, they would already have the information at hand. This will lead to a more efficient workflow around all points of the organization, which could potentially also increase profit revenue.

Data Summary

Per the data source that will be used, we will use a data set that has already been published for this specific area. We obtained this through this link: https://www.kaggle.com/datasets/shivachandel/kc-house-data. We will first clean the data and develop an ML model that can effectively predict house prices for the inquiring customer.

The data source we have obtained is in a CSV file format. After obtaining this data we will clean the data by dropping any unneeded columns, missing values, and outliers that may affect the way the data is compiled. The main columns that will be kept will be the main features that homebuyers and sellers are most interested in, such as the bedrooms, bathrooms, square footage, and the area they are looking to move to.

Once this has been accomplished, we will then implement an ML model that can train and test the data to make accurate decisions on home prices. This data will be more than sufficient to make a model. It contains all the features customers are looking for when they purchase a home. As far as any legal or ethical concerns, there should be none to worry about.

Implementation

The methodology used for building this machine learning model is going to be CRISP-DM. Using this methodology, we are more likely to see the best results for our project since we are needing to recognize the customer's needs. The following is what our implementation plan will look like using the CRISP-DM method:

- **Business Understanding –** Since we are dealing with customers' needs, we need to understand what they need. In this case, we are looking to predict the estimated cost of a home that encompasses the features the customer is looking for. By understanding and analyzing the customers' needs we can then research and implement machine learning techniques that can be used. The result of this step is that we can develop a plan on how to implement what we know.
- **Data Understanding –** To understand the full spectrum of what we are dealing with we also need to understand the data we are going to use and the relationships between it. By looking at the data we can identify outliers, null values, and fields that will not be necessary to our machine learning development. The result of this will give us clean valuable data we can implement in our model.
- **Data Preparation –** After understanding the data we have; we will need to prep it. We will need to exclude any column that will be of no use and delete any data that may contain outliers and null values. We can use a tool called Tableau to transform the data to where we need it to be, or we can directly clean it in the IDE. Also, we will need to check the data types to see if they need to be converted. The result of doing this will be that we have clean reliable data that is not tainted in any way.
- **Modeling –** After getting past the data prep we can now explore different machine learning models. We will be using a supervised learning technique to utilize the data labels. We are leaning toward a regression type of learning to give us accurate results. We can then split our data into 80% training and 20% test sets. The result of this will be finding the best model we can use to accurately predict a price.
- **Evaluation –** After understanding, prepping, and modeling the data we can look at our values more carefully to see if our predictions from the model are accurate. To check this, we would need to go back and analyze the requirements we established in the previous steps. If the model is deemed to be a failure, we will go back to the previous steps and reevaluate what needs to change before testing the model again. If the test passes, we will document and proceed to the last phase, deployment.
- **Deployment –** On this last phase, as the model is completed, we can start documenting our final report on the model about what we found and how it affected certain parts. For this model we can focus on giving the customer a UI to access this data. Then test again

to ensure no major bugs or requirements are missing. The result of this will be a successful application that can take user parameters and make a prediction.

Timeline

| Milestone or deliverable | Duration (hours or days) | Projected start date | Anticipated end date |
|---|---|---|---|
| Planning project and objectives | 6 days | 10/5/2024 | 10/11/2024 |
| Understand business goal | 4 days | 10/11/2024 | 10/15/2024 |
| Collect and compile data | 4 days | 10/15/2024 | 10/19/2024 |
| Data preparation (ETL) | 8 days | 10/19/2024 | 10/27/2024 |
| Evaluate different ML models | 7 days | 10/27/2024 | 11/3/2024 |
| Training and testing ML model | 5 days | 11/3/2024 | 11/8/2024 |
| Create UI for customer use | 8 days | 11/8/2024 | 11/16/2024 |
| Final documentation and testing | 3 days | 11/16/2024 | 11/19/2024 |
| Deploy successful ML model | 1 days | 11/19/2024 | 11/20/2024 |

| | | | |
|---|---|---|---|
| Ensure ML model/<br><br>data are maintained | Continuous | 11/20/2024 | --/--/---- |

Evaluation Plan

Here I will explain the verification methods we will use to ensure everything working properly. During the first phase, we will be simply getting an understanding of what we need to do and what we are looking to improve. Everyone involved in the understanding phase (developers, stakeholders, etc.) will be taking part in these discussions. Once we have agreed on a common goal and what the deliverable should look like we can have a solid planned foundation to build on from there.

The next step we will be looking at is obtaining the data for our models. During the data prep stage, we will convert the data into a viable sheet that we can use. By cleaning the data, we can data rid of any 'bad' data which includes outliers, missing values, or columns we may not need in general. This will allow our data to only house the field we need and nothing more.

The next verification method we will use for the next phase is the developing a model and testing phase. After exploring different model options, we should be set on which model will have the best performance with the data we have. In this case, we are more in need of accuracy by calculating the MAE (Mean Absolut Value) and the R squared value as our performance and accuracy tester. Verifying if the model work will be up to all involved. Communication between team members and the stakeholders will be essential to keep our goals and expectations in line. The code will be reviewed by different team members to ensure a different perspective in how this could be improved or implemented in a better way.

Before deployment, we will need to create a user-friendly interface so we can test it before releasing it. Ensuring that there are minimal bugs is necessary so to verify functionality, our software QA will unit test using different test cases. After a successful testing phase, we will do another testing phase with everyone involved with the project to ensure our goals and business objectives have been thoroughly met before release.

Resources and Costs

| Resources | Description | Cost |
|---|---|---|

| | | |
|---|---|---|
| Real estate agent | Will gather information about specific house info | Provided |
| Software QA | Responsible for testing the code | $2,000 |
| Project Manager | Planning and design of project | $10,000 |
| Machine Learning Engineer | Will be responsible for ML algorithm research, development and implementation | $9,000 |
| Data Scientist | Gathers data, exploration, transformation. | $5,000 |
| Programmer | Will develop user interface that customers will be using | 6,000 |
| Computers (5 total) | Used for development of application | $5,000 |

| | | |
|---|---|---|
| Customer | Participants that will gather feedback on how the new application feels | Provided |
| Software/libraries | Will be developed using Python 3 (all libraries needed) | ----- |
| Total Cost | ----- | $37,000 |

**Part C: Application**

The application is included in the submission as an .ipynb file. The dataset used will also be included.

**Part D: Post-implementation Report**

Solution Summary

As the housing market continues to boom, we have come up with a solution to provide customers with a self-serve home price predictor to avoid having to have real estate agents manually dig up numbers. This tool will allow customers to plug in their own features of what they are looking for in a home and get an immediate estimate on how much they are either looking at spending or how much their home will be worth. This allows for the organization to focus on other priorities that must be tended to and allows for a more positive and efficient workflow for the company as a whole. We have developed a Machine Learning model that utilizes *RandomForestRegression* and is able to take the data from all 20,000 homes that are for sale in King County, Washington State area and can give us a predicted price on a home that person is looking for by taking the features, such as bedrooms, bathrooms, square footage, and area, to give a predicted price.

Data Summary

- Provide the source of the raw data, how the data was collected, or how it was simulated.
- Describe how data was processed and managed throughout the application development life cycle: design, development, maintenance, or others.

We have obtained our dataset from Kaggle.com, it can be reached at: https://www.kaggle.com/datasets/shivachandel/kc-house-data.

**kc_house_data.csv** (2.38 MB)

Detail | Compact | Column | 10 of 21 columns

| id | date | price | bedrooms | bathrooms | sqft_living | sqft_lot |
|---|---|---|---|---|---|---|
| 1.00m – 9.90b | 372 unique values | 75k – 7.7m | 0 – 33 | 0 – 8 | 290 – 13.5k | 520 |
| 7129300520 | 20141013T000000 | 221900 | 3 | 1 | 1180 | 5650 |
| 6414100192 | 20141209T000000 | 538000 | 3 | 2.25 | 2570 | 7242 |
| 5631500400 | 20150225T000000 | 180000 | 2 | 1 | 770 | 10000 |
| 2487200875 | 20141209T000000 | 604000 | 4 | 3 | 1960 | 5000 |
| 1954400510 | 20150218T000000 | 510000 | 3 | 2 | 1680 | 8080 |
| 7237550310 | 20140512T000000 | 1.225E+06 | 4 | 4.5 | 5420 | 101930 |
| 1321400060 | 20140627T000000 | 257500 | 3 | 2.25 | 1715 | 6819 |
| 2008000270 | 20150115T000000 | 291850 | 3 | 1.5 | 1060 | 9711 |
| 2414600126 | 20150415T000000 | 229500 | 3 | 1 | 1780 | 7470 |
| 3793500160 | 20150312T000000 | 323000 | 3 | 2.5 | 1890 | 6560 |
| 1736800520 | 20150403T000000 | 662500 | 3 | 2.5 | 3560 | 9796 |
| 9212900260 | 20140527T000000 | 468000 | 2 | 1 | 1160 | 6000 |
| 114101516 | 20140528T000000 | 310000 | 3 | 1 | 1430 | 19901 |
| 6054650070 | 20141007T000000 | 400000 | 3 | 1.75 | 1370 | 9680 |

The original dataset had 21 columns, many more than what we needed. There were about 21,000 rows of housing data before the preprocess was started. All houses are located within the same county, King County, Washington State, USA.

| | A | B | C | D | E | F | G | H | I | J | K | L | M | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | id | date | price | bedrooms | bathrooms | sqft_living | sqft_lot | floors | waterfront | view | condition | grade | sqft_above | sqft |
| 2 | 7.13E+09 | 20141013T | 221900 | 3 | 1 | 1180 | 5650 | 1 | 0 | 0 | 3 | 7 | 1180 | |
| 3 | 6.41E+09 | 20141209T | 538000 | 3 | 2.25 | 2570 | 7242 | 2 | 0 | 0 | 3 | 7 | 2170 | |
| 4 | 5.63E+09 | 20150225T | 180000 | 2 | 1 | 770 | 10000 | 1 | 0 | 0 | 3 | 6 | 770 | |
| 5 | 2.49E+09 | 20141209T | 604000 | 4 | 3 | 1960 | 5000 | 1 | 0 | 0 | 5 | 7 | 1050 | |
| 6 | 1.95E+09 | 20150218T | 510000 | 3 | 2 | 1680 | 8080 | 1 | 0 | 0 | 3 | 8 | 1680 | |
| 7 | 7.24E+09 | 20140512T | 1.225e+00 | 4 | 4.5 | 5420 | 101930 | 1 | 0 | 0 | 3 | 11 | 3890 | |
| 8 | 1.32E+09 | 20140627T | 257500 | 3 | 2.25 | 1715 | 6819 | 2 | 0 | 0 | 3 | 7 | 1715 | |
| 9 | 2.01E+09 | 20150115T | 291850 | 3 | 1.5 | 1060 | 9711 | 1 | 0 | 0 | 3 | 7 | 1060 | |
| 10 | 2.41E+09 | 20150415T | 229500 | 3 | 1 | 1780 | 7470 | 1 | 0 | 0 | 3 | 7 | 1050 | |
| 11 | 3.79E+09 | 20150312T | 323000 | 3 | 2.5 | 1890 | 6560 | 2 | 0 | 0 | 3 | 7 | 1890 | |
| 12 | 1.74E+09 | 20150403T | 662500 | 3 | 2.5 | 3560 | 9796 | 1 | 0 | 0 | 3 | 8 | 1860 | |
| 13 | 9.21E+09 | 20140527T | 468000 | 2 | 1 | 1160 | 6000 | 1 | 0 | 0 | 4 | 7 | 860 | |
| 14 | 01141015 | 20140528T | 310000 | 3 | 1 | 1430 | 19901 | 1.5 | 0 | 0 | 4 | 7 | 1430 | |
| 15 | 6.05E+09 | 20141007T | 400000 | 3 | 1.75 | 1370 | 9680 | 1 | 0 | 0 | 4 | 7 | 1370 | |
| 16 | 1.18E+09 | 20150312T | 530000 | 5 | 2 | 1810 | 4850 | 1.5 | 0 | 0 | 3 | 7 | 1810 | |
| 17 | 9.3E+09 | 20150124T | 650000 | 4 | 3 | 2950 | 5000 | 2 | 0 | 3 | 3 | 9 | 1980 | |
| 18 | 1.88E+09 | 20140731T | 395000 | 3 | 2 | 1890 | 14040 | 2 | 0 | 0 | 3 | 7 | 1890 | |
| 19 | 6.87E+09 | 20140529T | 485000 | 4 | 1 | 1600 | 4300 | 1.5 | 0 | 0 | 4 | 7 | 1600 | |
| 20 | 00160003 | 20141205T | 189000 | 2 | 1 | 1200 | 9850 | 1 | 0 | 0 | 4 | 7 | 1200 | |
| 21 | 7.98E+09 | 20150424T | 230000 | 3 | 1 | 1250 | 9774 | 1 | 0 | 0 | 4 | 7 | 1250 | |
| 22 | 6.3E+09 | 20140514T | 385000 | 4 | 1.75 | 1620 | 4980 | 1 | 0 | 0 | 4 | 7 | 860 | |
| 23 | 2.52E+09 | 20140826T | 2e+006 | 3 | 2.75 | 3050 | 44867 | 1 | 0 | 4 | 3 | 9 | 2330 | |
| 24 | 7.14E+09 | 20140703T | 285000 | 5 | 2.5 | 2270 | 6300 | 2 | 0 | 0 | 3 | 8 | 2270 | |
| 25 | 8.09E+09 | 20140516T | 252700 | 2 | 1.5 | 1070 | 9643 | 1 | 0 | 0 | 3 | 7 | 1070 | |
| 26 | 3.81E+09 | 20141120T | 329000 | 3 | 2.25 | 2450 | 6500 | 2 | 0 | 0 | 4 | 8 | 2450 | |
| 27 | 1.2E+09 | 20141103T | 233000 | 3 | 2 | 1710 | 4697 | 1.5 | 0 | 0 | 5 | 6 | 1710 | |
| 28 | 1.79E+09 | 20140626T | 937000 | 3 | 1.75 | 2450 | 2691 | 2 | 0 | 0 | 3 | 8 | 1750 | |
| 29 | 3.3E+09 | 20141201T | 667000 | 3 | 1 | 1400 | 1581 | 1.5 | 0 | 0 | 5 | 8 | 1400 | |
| 30 | 5.1E+09 | 20140624T | 438000 | 3 | 1.75 | 1520 | 6380 | 1 | 0 | 0 | 3 | 7 | 790 | |

This is what the data looks like within the Excel spreadsheet. As you can see there will be a good amount of preprocessing data and getting rid of outliers.

By using Jupyter Notebooks, we are able to load our CSV file and we can start to clean the data. We first started dropping any unneeded columns besides 6, being: bedrooms, bathrooms, square footage, price, square foot lot, and zip code. Our data set will eventually look like this inside of Jupyter Notebook.

## Machine Learning

The machine learning method used for this particular application is RandomForest Regression. This specific algorithm uses ensemble learning that combines the predictions from multiple models to create and make a more accurate prediction. This method is very versatile as we can make the target variable ‘price’ and add features such as bedrooms, bathrooms, square footage, and zip code to make an accurate prediction on what the home may sell for or how much a house could be worth at that moment.

The way this method was developed is that we have a dataset for houses in the King County, Washington State area. We use Python to manipulate the data so that it’s easier to manage and make more accurate predictions without having columns that are irrelevant to the model itself. We use pandas to drop columns, get rid of null values, and look for outliers that might affect the results of the algorithm. We can then start to explore this data to see the

relationships between different features which will be shown later. After finding the right fit of features, we can then develop a model that can train, test, and split the data to make predictive results. To check the accuracy of our model we can implement the MAE score and the R-squared score.

This model was a desirable choice because it is very flexible to work with. We can experiment with feature engineering to find the best-suited features that will give us the most accurate results. Different combinations of features can give a better output for our results.

Validation

To score the validation of our model we calculated the MAE and the R2 scores which are displayed below. By using the scikitlearn here are the results of accuracy.

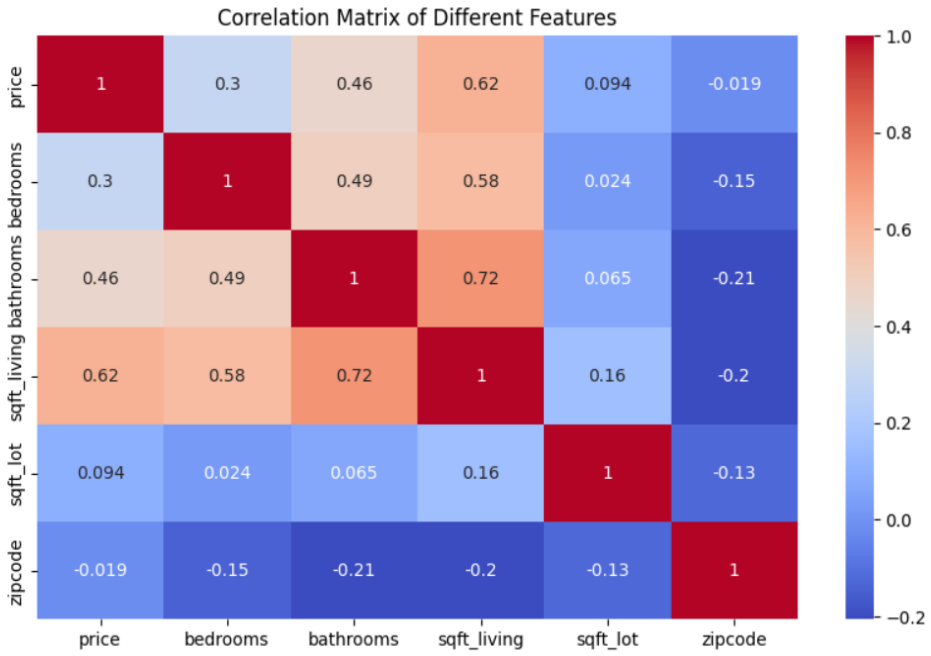
```
# MAE, MSE, & R2 validation for accuracy
mae = mean_absolute_error(y_test, y_pred)
print(f"Mean Absolute Error: {mae:.2f}")

mse = mean_squared_error(y_test, y_pred)
print(f"Mean Squared Error: {mse}")

r2 = r2_score(y_test, y_pred)
print(f"R² Score: {r2}")
```

```
Mean Absolute Error: 70964.57
Mean Squared Error: 10335416250.201336
R² Score: 0.7518152951521855
```

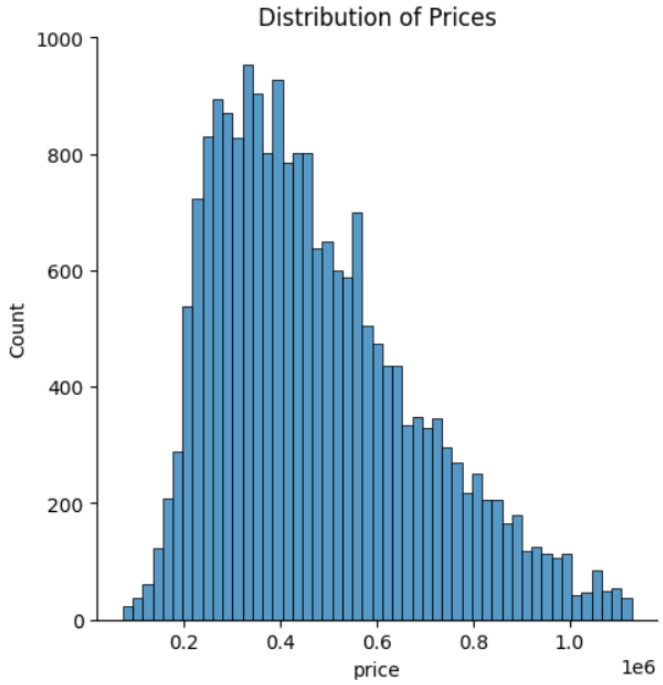
Although our accuracy result could be better. Going through the data, I noticed there were outliers in the price column. Before getting rid of the outliers the R2 score was about .70. After getting rid of the outliers, our score raised an additional .05 making it .7518. Although these are our results. There are things we could experiment with to raise the R2 scores and lower the MAE. We could experiment with hyperparameter tuning, looking for different features to combine (feature engineering) or adding additional data to the dataset for the model to train more data for better results.

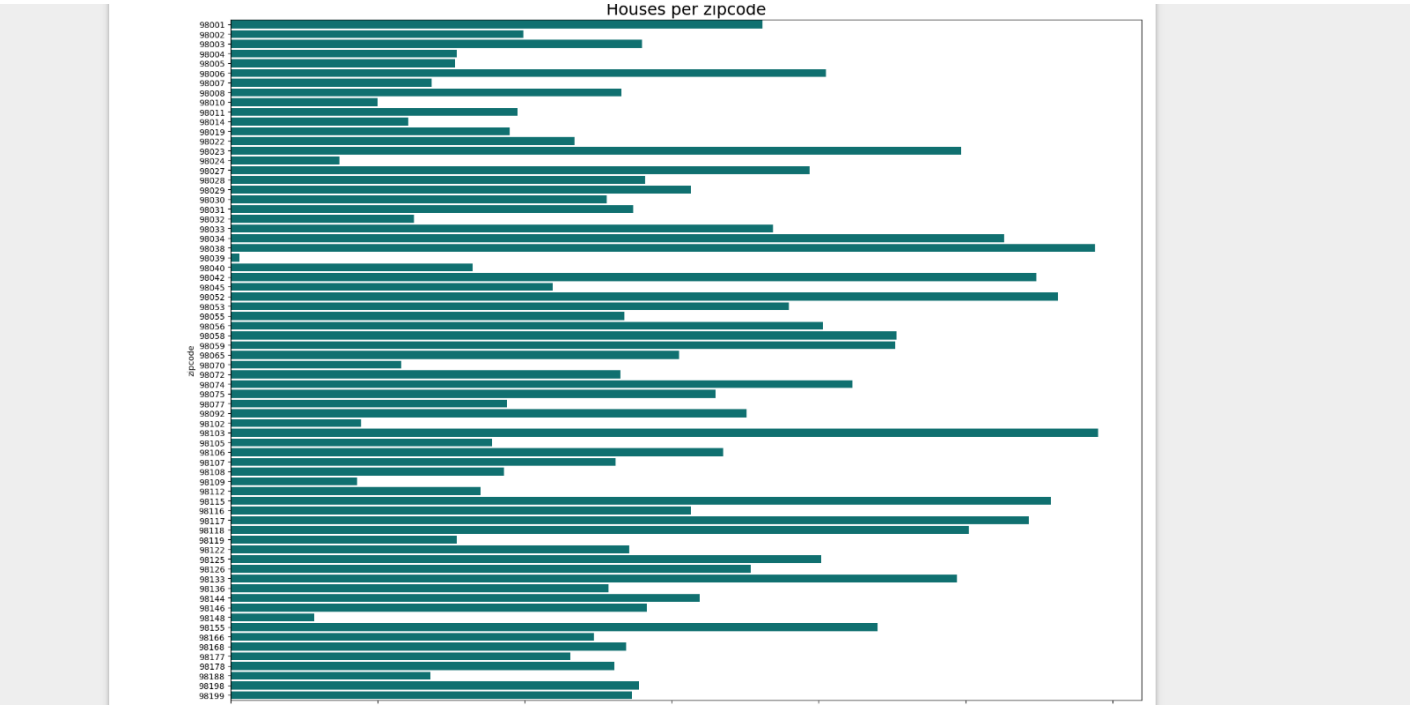
Visualizations

These are 3 visualizations that will also be generated within the notebook.




<Figure size 1000x600 with 0 Axes>



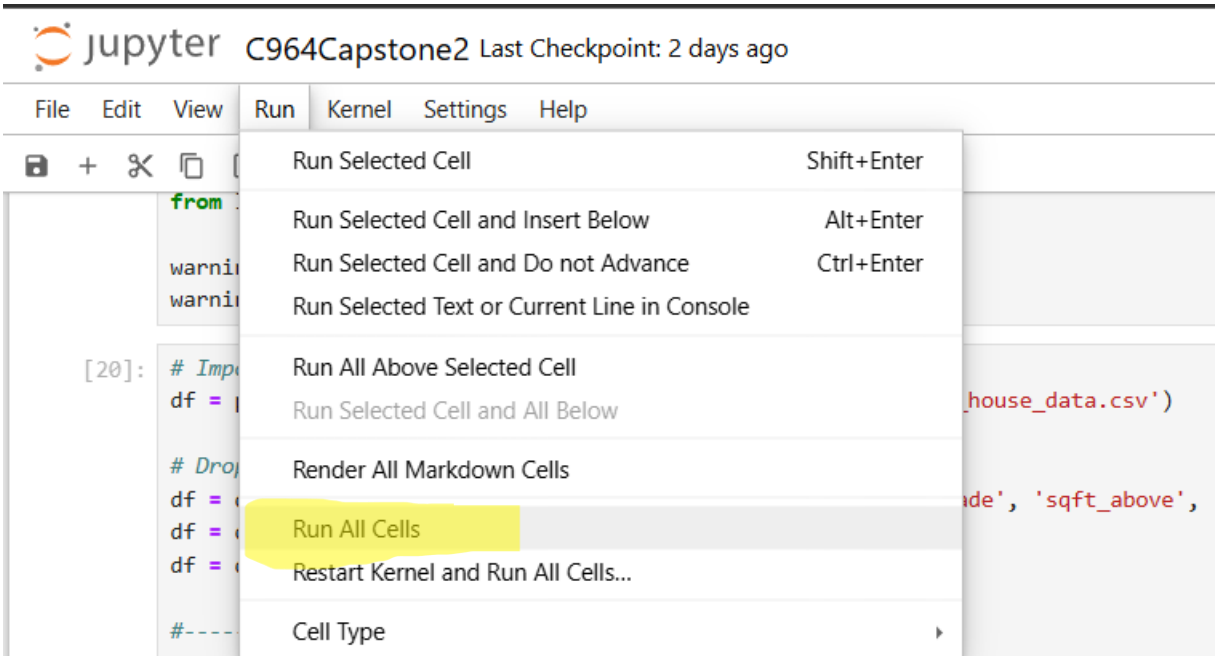



User Guide

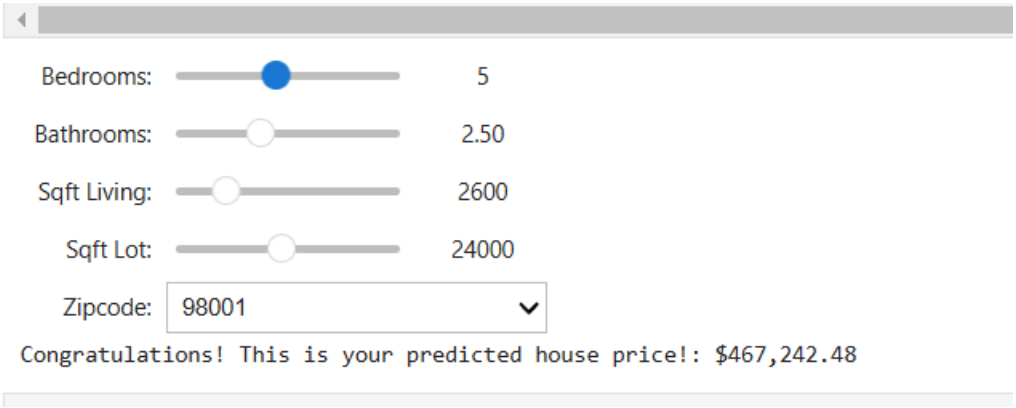
1. The files needed are all located in the same folder. When running the application be sure to change to file path to the correct path.

```
# Import dataset
df = pd.read_csv('/Users/iragl/Documents/WGU/C964Capstone2/kc_house_data.csv')

# Dropping columns, null values and fill N/A values
df = df.drop(columns=['id', 'date', 'waterfront', 'view', 'grade', 'sqft_above', 'sqft_basement', 'yr_built', 'yr_renova
df = df.dropna(subset=['price'])
df = df.fillna(method='ffill')

#---------------------------

# Definging a function to remove outliers in the price column
def remove_outliers(df, price):
    # Calculate Q1 & Q3
```

2. After changing the file path click the ‘Run All’ option.




3. The application will take about 10 seconds to execute. Afterwards, there will be a UI that appears at the end where you can input features of a house you are looking for. This will also give you live results instead of having to press a ‘predict’ button.



## Reference Page