Housing Price Prediction

Tomy Li He

C964 Computer Science Capstone

Table of Contents

A1. Letter of Transmittal

Tom Trident

Tom’s Real Estate

1234 T St

San Jose, CA 95112

Dear Tom,

In recent years, the housing market has experienced fluctuations in prices due to many competitors, interest rates, and government policies. These factors normally affect the housing market across the country and are unable to be foreseen in advance. However, your company can better prepare for changes in housing prices by considering factors that we can use to make predictions. A solution that can help your company attract more clients than other competitors is to implement a housing prediction system. This can provide your clients with a better understanding of the housing market in order to feel more secure when purchasing a home.

The housing prediction system will provide data visualizations that will benefit both the company and customers. In addition, the system will include housing price prediction capabilities given user input for specific house variables. It will help in making company business decisions as well as helping customers make purchases. Analysis of the data will provide the company with a better estimate of the housing prices depending on the area/neighborhood and which types of homes. This greater certainty will attract more customers and increase the revenue which could help in improving the system even more.

The funding needed to create the housing prediction system will be about $15,000. A majority of the funding will go towards the developmental team and the tools needed to accomplish tasks. Our team is well-versed in Python and working together for other similar projects. We are confident in our team’s skill and chemistry to provide your company with the system.

Thank you for your time and we look forward to hearing back from you soon.

Sincerely,

Tomy Li He

A2. Project Proposal

Problem Summary

Tom’s Real Estate is in a market that is saturated with other Real estate agencies. Furthermore, the housing market is in an uncertain state regarding housing prices. As a result, Tom’s Real Estate has been seeing fewer clients. However, adding the housing price prediction system will help the company as well as the clients in better understanding housing price patterns

Product Benefit

Tom’s Real estate attracts most of its customers through billboard, magazines, and newspaper advertisement. However, other competitors have the same exact approach. Therefore, by limiting client attraction through a saturated and archaic approach, it is difficult to increase the client base.

Our proposed housing price prediction solution will introduce greater certainty when predicting housing prices and convincing clients that the house is at a good price. The housing price prediction system will analyze a variety of factors in order to determine trends in housing prices. This information can be used as leverage when assisting clients in purchasing a new home. Another benefit of this new system is that it will provide the company with more information on the trends in housing prices which would put the company in a better position when making business decisions.

Product Outline

The product will be developed using Python and Jupyter Notebook. The program was written directly in Jupyter Notebook which allows code to be written and executed in separate cells. Python was determined to be the best language for this program because of its robust data analysis libraries and resources. The program will be able to take in housing cost factors such as garage area, total area of the house, number of cars that can fit in the garage, etc. to predict the cost of the house. This prediction will give customers a better sense of how much their desired home will cost. In addition, the data product will include data visualizations that can provide stakeholders and customers with information regarding the housing market.

Data Description

Our dataset for the housing price prediction system was retrieved from Kaggle.com. The dataset has information on 79 variables of residential homes in Iowa. The datasets are stored into 2 csv files. One of the two files represent the training dataset and the other represents the testing dataset. The data will be processed and normalized to run our algorithm on it. Listed below is a brief description of the data obtained from Kaggle:

- SalePrice - the property's sale price in dollars. This is the target variable that you're trying to predict.
- MSSubClass: The building class
- MSZoning: The general zoning classification
- LotFrontage: Linear feet of street connected to property
- LotArea: Lot size in square feet
- Street: Type of road access
- Alley: Type of alley access
- LotShape: General shape of property
- LandContour: Flatness of the property
- Utilities: Type of utilities available
- LotConfig: Lot configuration
- LandSlope: Slope of property
- Neighborhood: Physical locations within Ames city limits
- Condition1: Proximity to main road or railroad
- Condition2: Proximity to main road or railroad (if a second is present)
- BldgType: Type of dwelling
- HouseStyle: Style of dwelling
- OverallQual: Overall material and finish quality
- OverallCond: Overall condition rating
- YearBuilt: Original construction date
- YearRemodAdd: Remodel date
- RoofStyle: Type of roof
- RoofMatl: Roof material
- Exterior1st: Exterior covering on house
- Exterior2nd: Exterior covering on house (if more than one material)
- MasVnrType: Masonry veneer type
- MasVnrArea: Masonry veneer area in square feet
- ExterQual: Exterior material quality
- ExterCond: Present condition of the material on the exterior
- Foundation: Type of foundation
- BsmtQual: Height of the basement
- BsmtCond: General condition of the basement
- BsmtExposure: Walkout or garden level basement walls
- BsmtFinType1: Quality of basement finished area
- BsmtFinSF1: Type 1 finished square feet
- BsmtFinType2: Quality of second finished area (if present)
- BsmtFinSF2: Type 2 finished square feet
- BsmtUnfSF: Unfinished square feet of basement area
- TotalBsmtSF: Total square feet of basement area
- Heating: Type of heating
- HeatingQC: Heating quality and condition

Further information regarding the description of the data can be found in the data_description.txt file in the C964 folder attached.

Objective and Hypothesis

The primary objective of the housing price prediction system is to increase the number of house sales by improving the certainty and confidence of clients when they are looking for homes. The new system hypothesizes that the housing price prediction will be accurate enough to provide clients with enough information to purchase a home.

Methodology

The development of the housing price recommendation system will follow the agile methodology. This methodology puts emphasis on continuous testing and iterative development. Furthermore, the agile methodology involves heavy collaboration between the development team and the customer. Feedback received from stakeholders and customers will help with improving the quality of the final product. The agile methodology will be split into 5 distinct phases. The first phase (Requirements) involves communicating with the customers to acquire the requirements for the product. The second phase is development, and this is where the product is developed based on the defined requirements. The third phase encapsulates testing of the product, conducting quality assurance tests to detect any errors and meet acceptance criteria. The fourth phase is delivery of the product. The final phase is using feedback from stakeholders to work on a new iteration of the product

Funding Requirements

Funding for this project will require $15,000. This will cover the pay for the developmental team, QA team and integration team. Costs that incur due to out-of-scope requirements will not be covered by this funding. However, if need be, additional funding can be added to cover the cost of the added requirements as well for the additional cost of paying the three teams.

Impact of Solution

The implementation of the new system will benefit both the clients and the stakeholders. Tom’s Real Estate stakeholders will be able to make more well-informed business decisions through the housing prediction system’s data analysis capabilities. On the other hand, clients will have more insight into the value of different houses which would give them more confidence when purchasing houses. Client satisfaction will increase, and they will be more likely to recommend Tom’s Real Estate to friends and relatives.

Data Handling Precautions

All data used in the housing prediction system can be accessed publicly and does not violate any federal or state regulations. Customer and employee information will not be stored in anyway in the housing prediction system. Because sensitive data is not used during the development of the housing prediction system, a username and password will not be necessary when accessing the housing prediction system. However, if implemented into Tom's Real Estate existing systems, the company will be responsible for any security measures involving data protection.

Developer's Expertise

The individuals working on the development team are mostly alumni of WGU's computer science program. The team is experienced with Python and some experience with machine learning. The team will be led by a more senior developer who will act as the scrum master. This team has worked together in many similar projects that also adopted the agile methodology. The team's chemistry will allow them to complete the housing price prediction system by the deadline.

B. Project Proposal for IT Professionals

Problem Statement

The development of the housing price prediction system is crucial in increasing client attraction and overall revenue for the company. The housing prediction system uses machine learning algorithms to predict housing prices given a variety of factors. This will provide clients with more information that will aid in their decision in purchasing a house by knowing which houses are priced correctly or are good deals. In addition, the data visualizations can be used by stakeholders to make business decisions based on the housing market.

Customer Summary

The customers will include all existing customers of Tom's Real Estate. The system can also benefit potential customers who are not currently buying a home but are interested. Customers want to purchase their dream home at a price that they are happy and comfortable with. The housing price prediction system will take in specific factors such as number of bedrooms and predict the price of houses. Customers need information and support in making purchasing decisions in the housing market. The new system will be able to provide the housing prices based on the customer's needs.

Existing Systems

There is currently no system in place at Tom’s Real Estate that can predict housing prices. However, they do have information on a lot of houses that are currently on the market. This data could possibly be included into future iterations of the housing prediction system. The system is written in Python so integration and adaptation can be done easily in the future if needed.

Data

The datasets used in the housing price prediction system were obtained from Kaggle.com (link: https://www.kaggle.com/competitions/house-prices-advanced-regression-techniques).

A brief description of the 79 variables can be found in Section A in the Data Description part. Variables that are highly correlated to housing price sales will be used in our housing price prediction algorithm. Any outlying variables or empty values will be cleansed in order to prevent any errors in our calculations. The variables used may differ when datasets from Tom’s Real Estate is used due to a difference in correlation between variables.

Project Methodology

As stated in the project proposal in the methodology portion of Section A, the project will follow agile methodology. Communication between the development team, quality assurance team, and stakeholders is essential in developing a system that meets all requirements and for detecting errors or problems that occur in any of the phases. The agile phases that this project will follow are:

A) **Requirements –** The development team will collaborate with stakeholders to define the requirements for the new housing price prediction system. As feedback returns, the requirements can be altered for next iterations of the system.

B) **Development –** The development team will work to create the housing price prediction system based on the defined requirements

C) **Testing –** The quality assurance team will conduct continuous testing to detect any errors in the system and usability. Testing plans will be implemented as well as the use of black and white box testing.

D) **Delivery –** After the completion of testing, we will deliver the prototype for acceptance testing. Only when the acceptance criteria have been met will the new system be ready to use by Tom’s Real Estate

E) **Feedback –** Feedback will be collected from stakeholders and customers. Important feedback will be used as a basis for the new requirements when developing the next iteration of the system.

Deliverables

The project deliverables are the housing price prediction system, dataset used in the system, source code, documentation, and the data visualizations. This also includes any material created during the developmental cycle for this product.

Implementation Plan

The implementation of the new system will follow agile methodology and consist of two phases, which are the rollout of a prototype and the final product. At the end of each of these phases, the deliverables will be rolled out to stakeholders and customers. There will be a user manual given as well as a training dataset for the system.

Development will involve continuous testing and communication between every team and stakeholders. Testing will be conducted by the quality assurance team using testing plans. Once completed, the system will be delivered to stakeholders for acceptance testing. Afterwards, the system can be deployed.

The main deliverables at the end of the project will be the housing price prediction system. The source code, documentation, datasets, and data visualizations will also be included at the end of the project.

Evaluation Plan

The agile methodology involves continuous testing to detect errors or requirements that are not met. This prevents an incomplete product from being deployed and ensures all requirements are met.

The evaluation plan will be based on how helpful the housing price prediction system it to customers that want to purchase homes. The housing price prediction system will be evaluated based on the mean squared error, mean absolute error and root mean squared error which we use to determine how accurate our predictions are. Furthermore, we will provide a customer satisfaction survey to determine user satisfaction and the user experience. The survey will also include feedback that can be used for future iterations of the system or as a measurement of the success of the housing price prediction system

Resources and Costs

Programming Environment:

Python 3.10, necessary python machine learning libraries, Python default libraries, Seaborn, NumPy, Pandas, Matplotlib and Jupyter notebook. The resources used are compatible with Windows, Mac, and Linux. The cost of these resources will be $0 because they are all free and open-sourced.

Environment Costs:

Most of the work can be completed remotely so there is no need for additional workstations and other hardware. The system can be implemented into Tom's Real Estate's existing website.

Human Resource Requirements

The main cost of developing this new system comes from paying the team members for their hours for development, testing, and deploying. The cost may be subject to change due to potential out-of-scope requirements or change in the speed of each phase of development.

1) Phase 1 – Development and roll out of a prototype of the housing price prediction system. This also includes all the testing of the prototype – 80 hours ($8,000)

2) Phase 2 – Integrating feedback from the prototype. Acceptance testing by stakeholders and deploying actual system after bug fixes and feature improvements – 70 hours ($7,000)

Timeline and Milestones

Event	Start Date	End Date	Work Hours	Dependencies	Resource Assignment
Kick off meeting	9/1/2022	9/2/2022	8	None	Development Team, QA Team, stakeholders
Phase 1 (Product development and design)	9/2/2022	9/9/2022	40	Kick off meeting	Development Team
Phase 1 (QA testing)	9/9/2022	9/15/2022	22	Phase 1	QA Team

				(Product development and design)	
End of Phase 1 (Delivery of prototype)	9/15/2022	9/16/2022	10	Phase 1 (QA testing)	Development team, stakeholders
Phase 2 (Integration of feedback)	9/16/2022	9/19/2022	30	End of Phase 1 (Delivery of prototype)	Development team, stakeholders
Phase 2 (Acceptance testing and bug fixes)	9/19/2022	9/22/2022	30	Phase 2 (Integration of feedback)	Development team, QA team, stakeholders
End of Phase 2 (Stakeholder signoff and delivery of product	9/22/2022	9/23/2022	10	Phase 2 (Acceptance testing and bug fixes)	Development team, stakeholders
Total			150		

D. Product Documentation

Business Requirements

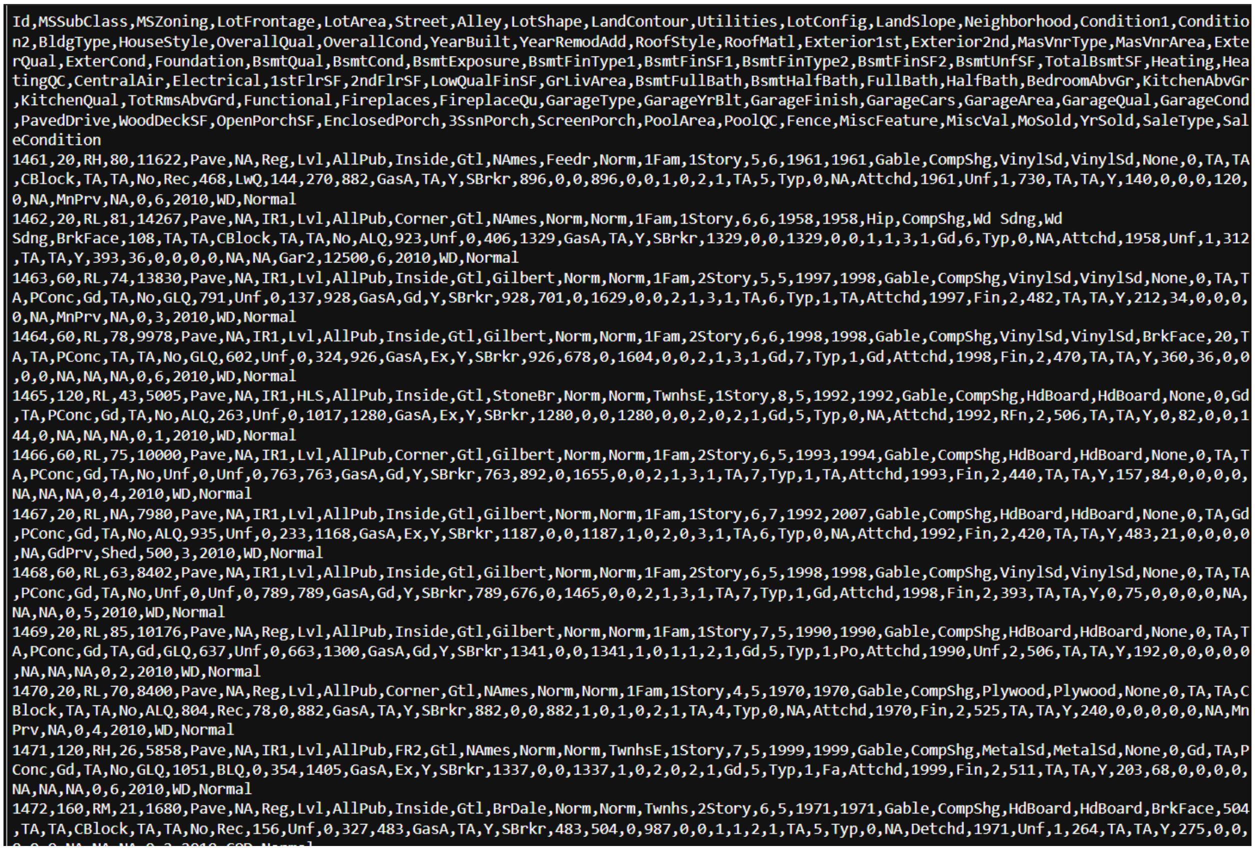
The business requirement of the new housing price prediction system was to use machine learning to predict house sales price based on multiple factors and to create data visualizations to help stakeholders and customers understand the housing market better.

The new system was developed to aid the company in providing customers with better information on houses and a better customer experience when purchasing or looking for potential homes. Because this new system uses machine learning and considers many different factors that can affect housing prices, it is able to predict housing prices more accurately which can provide customers with a confident estimate of a specific house. In addition, the data visualizations created by the system can be used by stakeholders as well as customers to make decisions based on predictions of the housing market.

Datasets

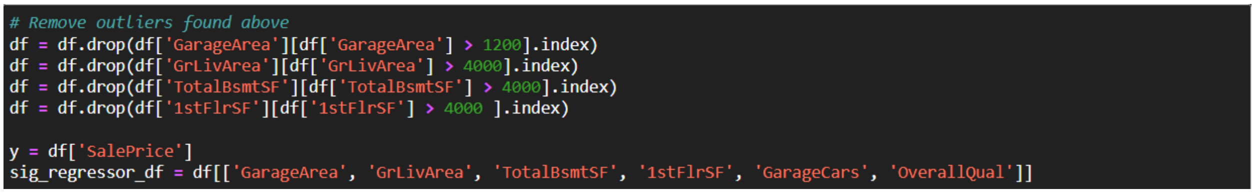
Datasets used for the housing price prediction system were obtained from Kaggle.com (Link: https://www.kaggle.com/competitions/house-prices-advanced-regression-techniques).

Datasets were stored in csv files. Figure 1 depicts an example of the raw dataset obtained from Kaggle.



```
Id,MSSubClass,MSZoning,LotFrontage,LotArea,Street,Alley,LotShape,LandContour,Utilities,LotConfig,LandSlope,Neighborhood,Condition1,Conditio
n2,BldgType,HouseStyle,OverallQual,OverallCond,YearBuilt,YearRemodAdd,RoofStyle,RoofMatl,Exterior1st,Exterior2nd,MasVnrType,MasVnrArea,Exte
rQual,ExterCond,Foundation,BsmtQual,BsmtCond,BsmtExposure,BsmtFinType1,BsmtFinSF1,BsmtFinType2,BsmtFinSF2,BsmtUnfSF,TotalBsmtSF,Heating,Hea
tingQC,CentralAir,Electrical,1stFlrSF,2ndFlrSF,LowQualFinSF,GrLivArea,BsmtFullBath,BsmtHalfBath,FullBath,HalfBath,BedroomAbvGr,KitchenAbvGr
,KitchenQual,TotRmsAbvGrd,Functional,Fireplaces,FireplaceQu,GarageType,GarageYrBlt,GarageFinish,GarageCars,GarageArea,GarageQual,GarageCond
,PavedDrive,WoodDeckSF,OpenPorchSF,EnclosedPorch,3SsnPorch,ScreenPorch,PoolArea,PoolQC,Fence,MiscFeature,MiscVal,MoSold,YrSold,SaleType,Sal
eCondition
1461,20,RH,80,11622,Pave,NA,Reg,Lvl,AllPub,Inside,Gtl,NAmes,Feedr,Norm,1Fam,1Story,5,6,1961,1961,Gable,CompShg,VinylSd,VinylSd,None,0,TA,TA
,CBlock,TA,TA,No,Rec,468,LwQ,144,270,882,GasA,TA,Y,SBrkr,896,0,0,896,0,0,1,0,2,1,TA,5,Typ,0,NA,Attchd,1961,Unf,1,730,TA,TA,Y,140,0,0,0,120,
0,NA,MnPrv,NA,0,6,2010,WD,Normal
1462,20,RL,81,14267,Pave,NA,IR1,Lvl,AllPub,Corner,Gtl,NAmes,Norm,Norm,1Fam,1Story,6,6,1958,1958,Hip,CompShg,Wd Sdng,Wd
Sdng,BrkFace,108,TA,TA,CBlock,TA,TA,No,ALQ,923,Unf,0,406,1329,GasA,TA,Y,SBrkr,1329,0,0,1329,0,0,1,1,3,1,Gd,6,Typ,0,NA,Attchd,1958,Unf,1,312
,TA,TA,Y,393,36,0,0,0,0,NA,NA,Gar2,12500,6,2010,WD,Normal
1463,60,RL,74,13830,Pave,NA,IR1,Lvl,AllPub,Inside,Gtl,Gilbert,Norm,Norm,1Fam,2Story,5,5,1997,1998,Gable,CompShg,VinylSd,VinylSd,None,0,TA,T
A,PConc,Gd,TA,No,GLQ,791,Unf,0,137,928,GasA,Gd,Y,SBrkr,928,701,0,1629,0,0,2,1,3,1,TA,6,Typ,1,TA,Attchd,1997,Fin,2,482,TA,TA,Y,212,34,0,0,0,
0,NA,MnPrv,NA,0,3,2010,WD,Normal
1464,60,RL,78,9978,Pave,NA,IR1,Lvl,AllPub,Inside,Gtl,Gilbert,Norm,Norm,1Fam,2Story,6,6,1998,1998,Gable,CompShg,VinylSd,VinylSd,BrkFace,20,T
A,TA,PConc,TA,TA,No,GLQ,602,Unf,0,324,926,GasA,Ex,Y,SBrkr,926,678,0,1604,0,0,2,1,3,1,Gd,7,Typ,1,Gd,Attchd,1998,Fin,2,470,TA,TA,Y,360,36,0,0
,0,0,NA,NA,NA,0,6,2010,WD,Normal
1465,120,RL,43,5005,Pave,NA,IR1,HLS,AllPub,Inside,Gtl,StoneBr,Norm,Norm,TwnhsE,1Story,8,5,1992,1992,Gable,CompShg,HdBoard,HdBoard,None,0,Gd
,TA,PConc,Gd,TA,No,ALQ,263,Unf,0,1017,1280,GasA,Ex,Y,SBrkr,1280,0,0,1280,0,0,2,0,2,1,Gd,5,Typ,0,NA,Attchd,1992,RFn,2,506,TA,TA,Y,0,82,0,0,1
44,0,NA,NA,NA,0,1,2010,WD,Normal
1466,60,RL,75,10000,Pave,NA,IR1,Lvl,AllPub,Corner,Gtl,Gilbert,Norm,Norm,1Fam,2Story,6,5,1993,1994,Gable,CompShg,HdBoard,HdBoard,None,0,TA,T
A,PConc,Gd,TA,No,Unf,0,Unf,0,763,763,GasA,Gd,Y,SBrkr,763,892,0,1655,0,0,2,1,3,1,TA,7,Typ,1,TA,Attchd,1993,Fin,2,440,TA,TA,Y,157,84,0,0,0,0,
NA,NA,NA,0,4,2010,WD,Normal
1467,20,RL,NA,7980,Pave,NA,IR1,Lvl,AllPub,Inside,Gtl,Gilbert,Norm,Norm,1Fam,1Story,6,7,1992,2007,Gable,CompShg,HdBoard,HdBoard,None,0,TA,Gd
,PConc,Gd,TA,No,ALQ,935,Unf,0,233,1168,GasA,Ex,Y,SBrkr,1187,0,0,1187,1,0,2,0,3,1,TA,6,Typ,0,NA,Attchd,1992,Fin,2,420,TA,TA,Y,483,21,0,0,0,0
,NA,GdPrv,Shed,500,3,2010,WD,Normal
1468,60,RL,63,8402,Pave,NA,IR1,Lvl,AllPub,Inside,Gtl,Gilbert,Norm,Norm,1Fam,2Story,6,5,1998,1998,Gable,CompShg,VinylSd,VinylSd,None,0,TA,TA
,PConc,Gd,TA,No,Unf,0,Unf,0,789,789,GasA,Gd,Y,SBrkr,789,676,0,1465,0,0,2,1,3,1,TA,7,Typ,1,Gd,Attchd,1998,Fin,2,393,TA,TA,Y,0,75,0,0,0,0,NA,
NA,NA,0,5,2010,WD,Normal
1469,20,RL,85,10176,Pave,NA,Reg,Lvl,AllPub,Inside,Gtl,Gilbert,Norm,Norm,1Fam,1Story,7,5,1990,1990,Gable,CompShg,HdBoard,HdBoard,None,0,TA,T
A,PConc,Gd,TA,Gd,GLQ,637,Unf,0,663,1300,GasA,Gd,Y,SBrkr,1341,0,0,1341,1,0,1,1,2,1,Gd,5,Typ,1,Po,Attchd,1990,Unf,2,506,TA,TA,Y,192,0,0,0,0,0
,NA,NA,NA,0,2,2010,WD,Normal
1470,20,RL,70,8400,Pave,NA,Reg,Lvl,AllPub,Corner,Gtl,NAmes,Norm,Norm,1Fam,1Story,4,5,1970,1970,Gable,CompShg,Plywood,Plywood,None,0,TA,TA,C
Block,TA,TA,No,ALQ,804,Rec,78,0,882,GasA,TA,Y,SBrkr,882,0,0,882,1,0,1,0,2,1,TA,4,Typ,0,NA,Attchd,1970,Fin,2,525,TA,TA,Y,240,0,0,0,0,0,NA,Mn
Prv,NA,0,4,2010,WD,Normal
1471,120,RH,26,5858,Pave,NA,IR1,Lvl,AllPub,FR2,Gtl,NAmes,Norm,Norm,TwnhsE,1Story,7,5,1999,1999,Gable,CompShg,MetalSd,MetalSd,None,0,Gd,TA,P
Conc,Gd,TA,No,GLQ,1051,BLQ,0,354,1405,GasA,Ex,Y,SBrkr,1337,0,0,1337,1,0,2,0,2,1,Gd,5,Typ,1,Fa,Attchd,1999,Fin,2,511,TA,TA,Y,203,68,0,0,0,0,
NA,NA,NA,0,6,2010,WD,Normal
1472,160,RM,21,1680,Pave,NA,Reg,Lvl,AllPub,Inside,Gtl,BrDale,Norm,Norm,Twnhs,2Story,6,5,1971,1971,Gable,CompShg,HdBoard,HdBoard,BrkFace,504
,TA,TA,CBlock,TA,TA,No,Rec,156,Unf,0,327,483,GasA,TA,Y,SBrkr,483,504,0,987,0,0,1,1,2,1,TA,5,Typ,0,NA,Detchd,1971,Unf,1,264,TA,TA,Y,275,0,0,
```

Figure 1. Raw test.csv file

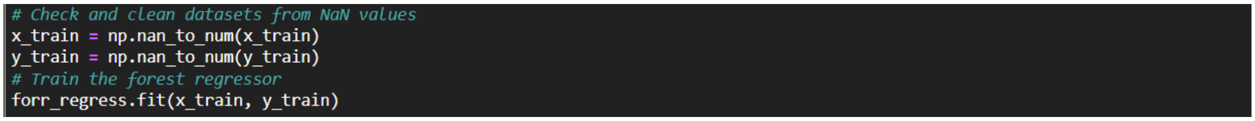


```
# Remove outliers found above
df = df.drop(df['GarageArea'][df['GarageArea'] > 1200].index)
df = df.drop(df['GrLivArea'][df['GrLivArea'] > 4000].index)
df = df.drop(df['TotalBsmtSF'][df['TotalBsmtSF'] > 4000].index)
df = df.drop(df['1stFlrSF'][df['1stFlrSF'] > 4000 ].index)

y = df['SalePrice']
sig_regressor_df = df[['GarageArea', 'GrLivArea', 'TotalBsmtSF', '1stFlrSF', 'GarageCars', 'OverallQual']]
```

Figure 2. Dropping outlying variables to determine the regressors

We determined which variables were highly correlated with sales price in order to drop to conduct random forest regression. Figure 2 shows that we dropped some variables from the training dataset.



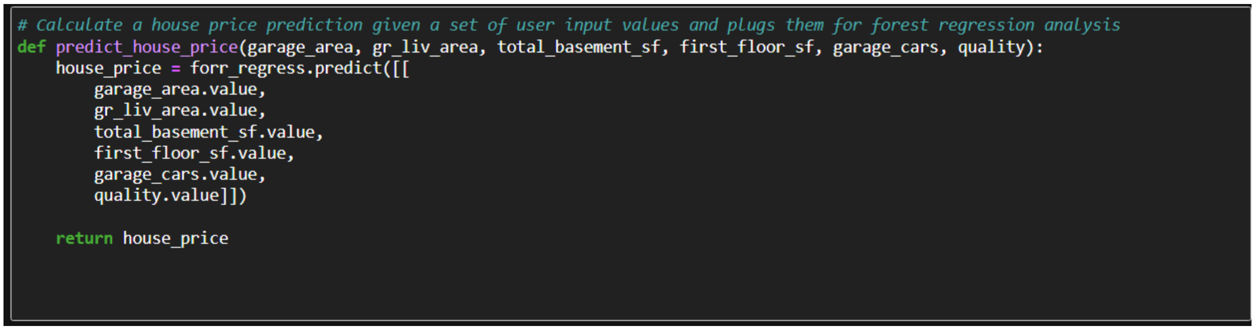
```
# Check and clean datasets from NaN values
x_train = np.nan_to_num(x_train)
y_train = np.nan_to_num(y_train)
# Train the forest regressor
forr_regress.fit(x_train, y_train)
```

Figure 3. Cleaning datasets for forest regression

The datasets were also checked and clean of NaN values before being used in the forest regression model.

Data Product Code

The housing prediction system involves the use of pandas, sklearn, seaborn, and NumPy in order to create a predictive algorithm for housing prices. Specifically, the main predictive function involves sklearn's RandomForestRegressor predict function to calculate the house price given the 6 variables listed in Figure 3.

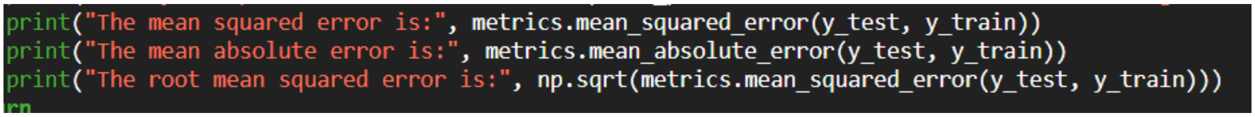


```
# Calculate a house price prediction given a set of user input values and plugs them for forest regression analysis
def predict_house_price(garage_area, gr_liv_area, total_basement_sf, first_floor_sf, garage_cars, quality):
    house_price = forr_regress.predict([[
        garage_area.value,
        gr_liv_area.value,
        total_basement_sf.value,
        first_floor_sf.value,
        garage_cars.value,
        quality.value]])

    return house_price
```

Figure 3. Predictive function

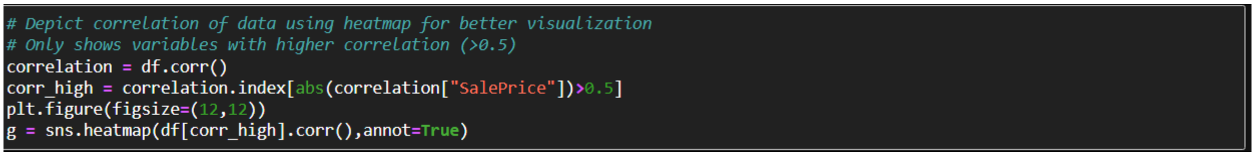
Furthermore, Figure 4 depicts the use of the metrics module from sklearn to calculate the mean squared error, mean absolute error, and root mean squared error as evaluation metrics for the prediction.



```
print("The mean squared error is:", metrics.mean_squared_error(y_test, y_train))
print("The mean absolute error is:", metrics.mean_absolute_error(y_test, y_train))
print("The root mean squared error is:", np.sqrt(metrics.mean_squared_error(y_test, y_train)))
rn
```

Figure 4. Calculations of evaluation metrics for the price prediction

The last sets of data product that are included in the project are the data visualizations. The descriptive functionality required the use of seaborn in order to produce a histogram, normalized histogram, and a heatmap to show the correlation of the data. Figure 5 shows an example of the code used to generate our heatmap data visualization.



```
# Depict correlation of data using heatmap for better visualization
# Only shows variables with higher correlation (>0.5)
correlation = df.corr()
corr_high = correlation.index[abs(correlation["SalePrice"])>0.5]
plt.figure(figsize=(12,12))
g = sns.heatmap(df[corr_high].corr(),annot=True)
```

Figure 5. Use of seaborn to generate heatmap

Hypothesis Verification

The hypothesis will be verified by analyzing the mean absolute error, mean squared error, and root mean squared error of our prediction to determine if the results are accurate enough. Evaluation metric values (MAE, MSE, RMSE) between 0.0 and 5.0 will suggest that the prediction is accurate based on the team's agreed criteria. More time will be needed to get accurate results regarding the effectiveness of the housing price prediction system because quarterly and annual reports will need to be compared to determine the difference in house sales before and after the implementation of the housing prediction system.

Effective Visualizations and Reporting

The data product includes data visualizations such as a histogram and normalized histogram showing the distribution of house sales based on sale prices, a heatmap that visualizes the correlations between different variables, and line plots that depict the relations between house prices and highly correlated variables.

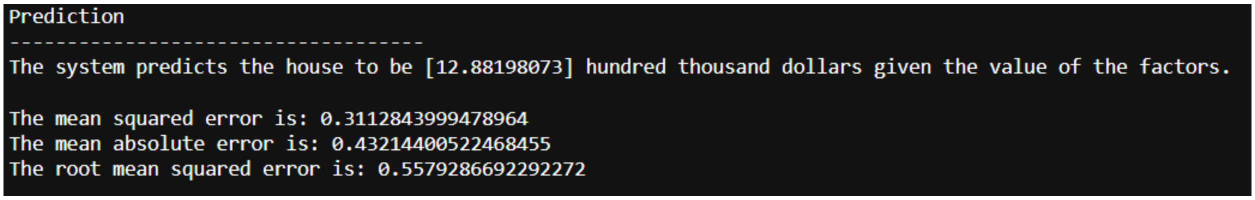
The main purpose of the histograms is to show the distribution of the number of houses purchased based on the price of the house. This information can provide stakeholders with insight on which price range home buyers are more likely to be interested in.

The heatmap visualization depicts the correlation between variables and helps with determining relationships between variables. For the data product heatmap, we utilized a correlation coefficient of >=0.5 to depict variables that have higher correlation with house sales price. This shortens the list of aspects that can have a greater effect on housing prices so that stakeholders can make more well-informed business decisions.

Lastly, the line plots are mainly used to serve as a visual depiction for the relation between the most highly correlated variables and house sales price.

Accuracy Analysis

The accuracy of the housing prediction system depends on the three evaluation metrics used which are the mean squared error, mean absolute error, and root mean squared error. Based on our criteria for accuracy, we want the MSE, MAE, and RMSE values to be as close to 0.0 as possible because we want to minimize error in the system's predictions. Evaluation metric values ranging from 0.0 to 5.0 will be considered accurate based on our team's criteria. This may not hold true for different datasets or other models, so this conclusion was made based on the team's own criteria. After completing the calculations, we determined that the evaluations suggest that our predictions are accurate. Figure 6 below shows the MSE, MAE, and RMSE calculated using sklearn modules.



```
Prediction
------------------------------------
The system predicts the house to be [12.88198073] hundred thousand dollars given the value of the factors.

The mean squared error is: 0.3112843999478964
The mean absolute error is: 0.43214400522468455
The root mean squared error is: 0.5579286692292272
```

Figure 6. Prediction price, MSE, MAE, and RMSE calculated from user input

However, more time is needed to determine the housing price prediction system's efficacy in increasing the number of house sales.

Application Testing

The development of the new housing price prediction system will involve continuous testing throughout the phases. Testing will follow the agile methodology. During the first phase, the prototype of the system will undergo functionality testing. Any bug fixes or changes will require regressing testing to be completed in order to ensure changes have not caused any other issues. In the second phase, acceptance test must be conducted by Tom's Real Estate to verify that all requirements have been implemented.

Application Files

The application files can be found in the C964 zip folder submitted with this document. The zip folder includes:

- HousingPricePrediction.ipynb – Jupyter notebook file of the housing price prediction system
- test.csv – the test dataset
- train.csv – the training dataset
- requirements.txt – text file that lists out all the packages needed to run the housing price prediction program.

User's guide

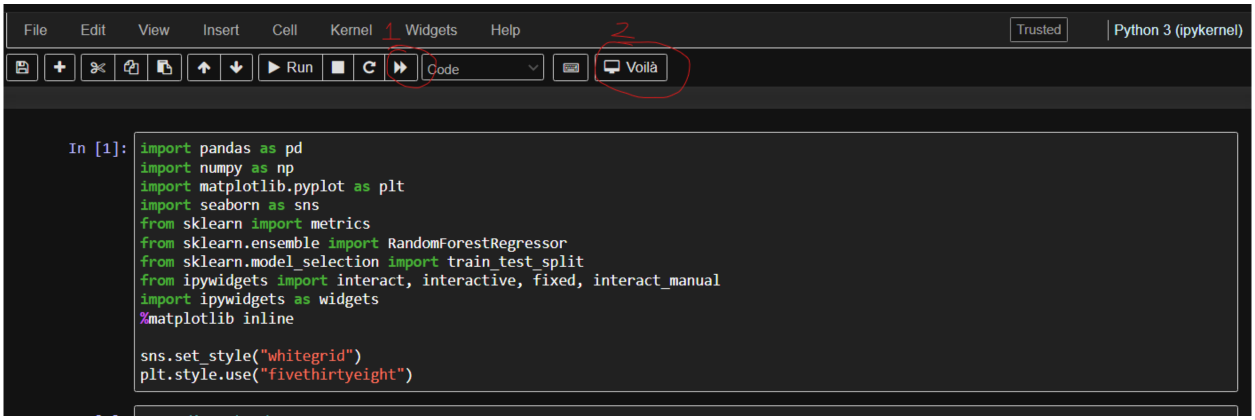
The application will need Jupyter notebook or any IDE that can run ipynb files.

1. Unzip the C964 folder and place the folder somewhere accessible.

2. Open requirements.txt and confirm that all necessary packages are installed.

3. Open the HousingPricePrediction.ipynb file.

4. Press the run all cells (Label 1) or Voila button (Label 2) to run the application. Figure 7 shows the location of both buttons.

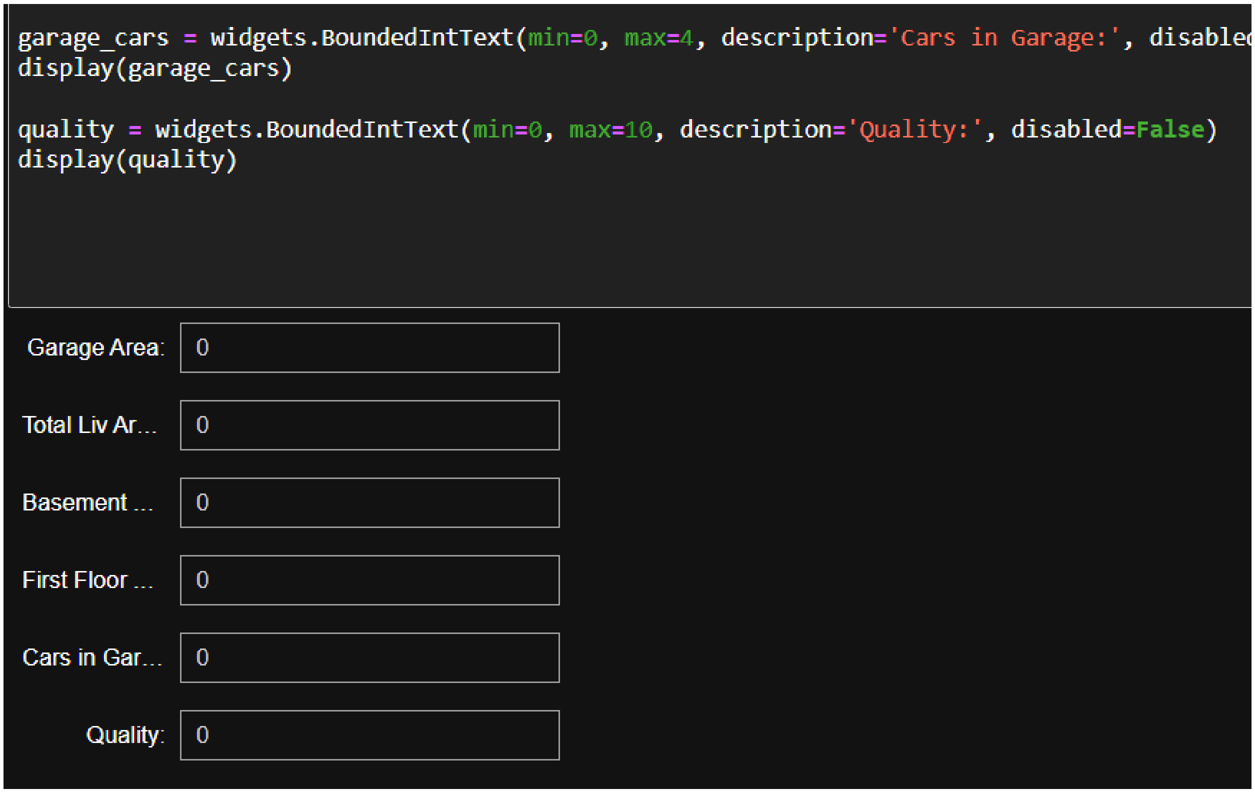


```
In [1]: import pandas as pd
        import numpy as np
        import matplotlib.pyplot as plt
        import seaborn as sns
        from sklearn import metrics
        from sklearn.ensemble import RandomForestRegressor
        from sklearn.model_selection import train_test_split
        from ipywidgets import interact, interactive, fixed, interact_manual
        import ipywidgets as widgets
        %matplotlib inline

        sns.set_style("whitegrid")
        plt.style.use("fivethirtyeight")
```

Figure 7. Example of run and voila button

5. Once the application has been run, there will be fields where you can enter in values for different variables that affect housing prices. Figure 8 below shows the fields. The fields are Garage Area, Total Living Area, Basement Area, First Floor Area, Cars in Garage, and Quality.



```
garage_cars = widgets.BoundedIntText(min=0, max=4, description='Cars in Garage:', disablec
display(garage_cars)

quality = widgets.BoundedIntText(min=0, max=10, description='Quality:', disabled=False)
display(quality)
```

Garage Area: 0

Total Liv Ar... 0

Basement ... 0

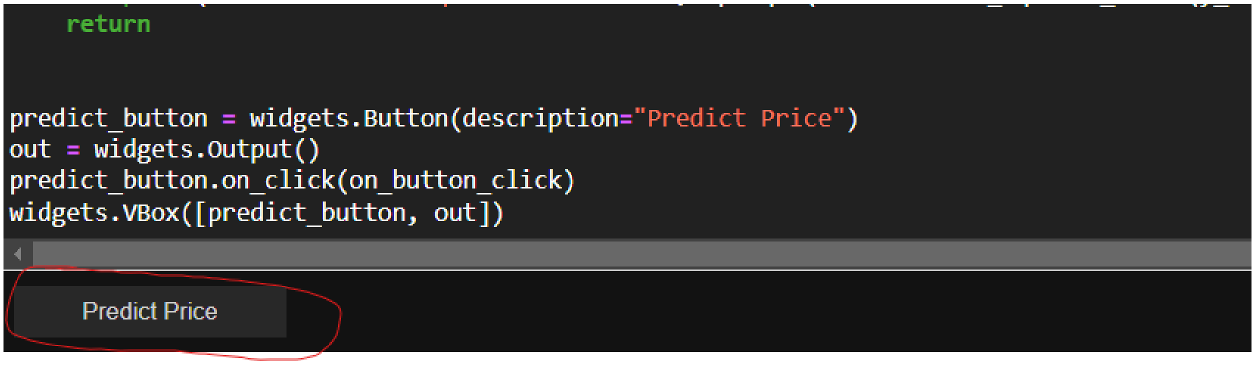
First Floor ... 0

Cars in Gar... 0

Quality: 0

Figure 8. User input text fields for variable values.

6. After filling out all the values in the fields. Press the Predict Price button the calculate the price prediction, MSE, MAE, and RMSE. The button is shown in Figure 9.



```
    return


predict_button = widgets.Button(description="Predict Price")
out = widgets.Output()
predict_button.on_click(on_button_click)
widgets.VBox([predict_button, out])
```

Figure 9. Example of the Predict Price button

Sources

House prices - advanced regression techniques. Kaggle. (n.d.). Retrieved September 15, 2022, from https://www.kaggle.com/competitions/house-prices-advanced-regression-techniques