

ML4641_Team1

ML4641 Team 1 Final Report

Taiki Aiba, Jeongyeop Han, Sean Liu, Arian Patel, Devin Zhang

1. Introduction/Background:

○ Literature Review:

- Extensive work has been conducted in housing price prediction. Park and Bae [1] successfully leveraged machine learning algorithms to help real estate agents list houses in Fairfax County, Virginia. Soltani [2] successfully created a model to predict housing prices in Adelaide, Australia. Mora-Garcia et al. [3] did the same for Alicante, Spain.
- However, this research is concentrated in housing price prediction, often targeted towards sellers maximizing profit instead of homebuyers. Housing affordability is still a loose concept [4] and the best variables to predict an affordability is still being studied [5]. Irregardless, housing affordability is a big factor for homebuyers, alongside accessibility, community income, household density, and employment access [6].
- Research in machine learning approaches to housing recommendation is limited. One such effort is conducted by Shi and Jiang [7], who extracted user's preference for housing sources and behavioral data to return personalized housing recommendations.

○ Datasets Used:

- "Rental Properties Collaboration Data," [www.kaggle.com](https://www.kaggle.com/datasets/arashnic/property-data).
<https://www.kaggle.com/datasets/arashnic/property-data> (accessed Apr. 20, 2024).
 - This dataset contains properties of a house (e.g. number of rooms, has elevator, has storage area, monthly rent) and prospective renter/buyer's responses.
- "Housing Prices Dataset," [www.kaggle.com](https://www.kaggle.com/datasets/yasserh/housing-prices-dataset/data).
<https://www.kaggle.com/datasets/yasserh/housing-prices-dataset/data> (accessed Apr. 20, 2024).
 - Contains price of a house and variables such as area, number of bedrooms and bathrooms, and special features it has

2. Problem Definition:

○ Problem:

- Housing is a concern for many. People, especially younger generations and those in expensive cities, often struggle with whether to buy or rent a house. Factors such as inflation and increasing mortgage rates compound the challenge of finding housing.
- Motivation:
 - Choosing the ideal house within financial constraints is important. Home buyers have specific needs and wants, such as proximity to work, school district, and house size. Thus, predicting prices, comparing house features, and deciding whether to buy or rent are essential.


3. Methods:

- Decision Tree (supervised):
 - Data Preprocessing:
 - The data used for this model came from a sheet of rental properties and associated activities from users, such as a new visit to the property, a successful meeting, and closing the deal. The goal is to understand what features of the property leads a user to closing the deal. The challenge from this dataset came from extremely messy cost data for deposit and monthly rent - values for rent and deposit ranged from 4 million to 7.2 million. As we don't know the currency or location of these properties, it's challenging to judge if the values are reasonable based on ground truth, so we opted to clean the data by assuming it was normally distributed and cutting any outliers beyond 3 standard deviations. These outliers are likely due to users adding dummy values.
 - Additional data pre-processing and cleaning included encoding categorical variables such as having an elevator or storage area, replacing missing values with the mode of that column, and turning the event types like deal-success into one-hot values.
 - Algorithm/Model:
 - From there, a decision tree was trained with the target on the "deal success" event type using TensorFlow's decision forest with an 80/20 training/validation split. We chose to use a decision tree because it would show the features that are most important in deciding whether to rent a house. For any given house, someone could use the decision tree to follow its features and ultimately determine whether to rent the house.
- Regression (supervised):
 - Data Preprocessing:
 - We converted all the categorical variables to numerical via one-hot encoding, and normalized the variables. Then we removed outliers using interquartile range, and used principal component analysis to reduce the number of dimensions.

- Algorithm/Model:
 - We used sklearn's linear, lasso, and logistic regression models, with an 80/20 training/validation split. We chose to use regression because it is meant for predicting the value of one variable given other variables. There are many different types of regression models, each with its own uses; we used linear regression because it is intuitive, lasso because it helps prevent overfitting, and logistic because it is robust to outliers.
- Clustering (unsupervised):
 - Data Preprocessing:
 - We visualized two different datasets to understand the distribution of the datasets' features. We decided to use the dataset whose values are more reasonable, because the other dataset had wild outliers. Specifically, the dataset we decided not to use had values such as 200 bathrooms or 120 bedrooms.
 - Then we used one-hot encoding to convert the categorical data to numerical values, and reduced the number of dimensions from 13 to 3 with forward feature selection.
 - Algorithm/Model:
 - We implemented DBSCAN, K-Means, and GMM using the sklearn library. Because we do not know how many clusters there actually are, we experimented with hard and soft clustering models, and tried different numbers of clusters. In DBSCAN, the criteria was a radius of 0.5 and minimum neighbor count of 5. For K-Means, we used the elbow method to find the optimal value of K, which was 2- though 3 and 4 also were not bad ideas. Then we implemented K-Means for K = 2, 3, and 4. Finally, for GMM, we used Bayesian information criterion (BIC) to determine the optimal number of clusters, which is 6; from there we implemented GMM with 6 clusters. In addition to implementing the models, we created visualizations for each model.

4. Results and Discussion:

- Decision Tree:
 - We first created a decision tree using all 5 features ("room_qty", "unit_area", "elevator", "building_floor_count", "storage"), resulting in a high accuracy of 0.993, and looks like the following. Please note that you need to open the image in a new tab to see it clearly.
 - These are the measures to evaluate the performance of our 5-feature decision tree.
 - Accuracy: 0.993
 - Precision: 0.981

- Recall: 0.99
- F1 Score: 0.985
- These are the boxplots of the the deposit and monthly rent, before and after cleaning.
Clean Data Visualization
- This model is very complex, so we sought to see if we could cut out any features. We used a recursive feature elimination technique, based on a stratified K-fold method. We were expecting that we would rapidly gain accuracy as we went from very 1 feature to 2-3 features, and then having the accuracy plateau. Instead, we found that accuracy remained nearly constant as we cut out features, shown in the figure below.
- From this, we could just use a single feature, in this case the “unit_area” of a dwelling to predict with nearly identical accuracy (99.28%) if someone will choose to rent the home or not. This simplified model is shown as follows.
- Please note that there are still many leaves on this tree, but this is due to the way that this visualization handles displaying other features - the final rent vs not rent decision is just on the unit area. This provides an indication of what realtors should look for as they offer places to rent, or if they choose to instead put that apartment up for sale.
- Regression:
 - These are the visualizations and accuracies of each regression model we implemented:
 - Linear regression:
 - root mean squared error = $5.1098045225437246e-08$
 - Lasso regression:
 - root mean squared error = $3.0215379169704954e-07$
 - Logistic regression:
 - root mean squared error = $3.0046382410075513e-07$
- Clustering:
 - These are the visualizations and accuracies of each clustering model we implemented.
 - DBSCAN accuracies:
 - Silhouette Score: -0.18707307274001764
 - Calinski-Harabasz Index: 6.182089437243131
 - Davies-Bouldin Index: 1.3571312867411092
 - K-Means:
 - K-Means with 2 clusters:
 - Silhouette Score: 0.19720861390886454
 - Calinski-Harabasz Index: 110.02377547399512
 - Davies-Bouldin Index: 2.129314981138082

- K-Means with 3 clusters:
 - Silhouette Score: 0.16150905135122215
 - Calinski-Harabasz Index: 87.0236352335191
 - Davies-Bouldin Index: 1.989455784259576
- K-Means with 4 clusters:
 - Silhouette Score: 0.17250978214823456
 - Calinski-Harabasz Index: 79.68900965832557
 - Davies-Bouldin Index: 1.9907834210480944
- GMM accuracies:
 - Silhouette Score: 0.08665692335927525
 - Calinski-Harabasz Index: 39.40906437336789
 - Davies-Bouldin Index: 3.0697599270489992
- These are the boxplots of the features in the dataset we used.
- Analysis of model:
 - Because we want to make silhouette score close to 1, maximize the Calinski-Harabasz index, and minimize Davies-Bouldin index, the best clustering model is K-Means with 2 clusters.
 - Clustering did not achieve as high accuracy as the decision tree and regression, likely because clustering, unlike the other two models, is unsupervised (i.e. no test data or known ground truths). The clusters are also not very well-defined, as shown by the low silhouette scores. However, the clustering still gave a good idea of house price levels and each level's area range.
- Comparison of Models:
 - Each model served a different purpose, but overall the decision tree achieved the best accuracy (but also has the most complex results/visualization).
 - Decision tree helps people decide whether to rent a house given its features.
 - Regression predicts house price and helps sellers determine a suitable price based on its features.
 - Clustering helps with home recommendation based on similar features, and gives an appropriate price range given house area.
 - Decision tree and regression were easier to train because they are supervised and we have the labels/known ground truths, whereas clustering (which is unsupervised) did not.
- Next Steps:
 - Improve accuracy of clustering to be on par with that of decision tree and regression
 - Create home recommendation system using multidimensional clustering

- Evaluate house (e.g. predict prices, decide whether to rent) based on its images and reviews, which integrate computer vision and natural language processing into our project

5. References

- [1] Park, Byeonghwa, and Jae Kwon Bae. "Using machine learning algorithms for housing price prediction: The case of Fairfax County, Virginia Housing Data." *Expert Systems with Applications*, vol. 42, no. 6, Apr. 2015, pp. 2928–2934, <https://doi.org/10.1016/j.eswa.2014.11.040>.
- [2] A. Soltani, M. Heydari, F. Aghaei, and C. J. Pettit, "Housing price prediction incorporating spatio-temporal dependency into machine learning algorithms," *Cities*, vol. 131, p. 103941, Dec. 2022. doi:10.1016/j.cities.2022.103941
- [3] R.-T. Mora-Garcia, M.-F. Cespedes-Lopez, and V. R. Perez-Sanchez, "Housing price prediction using machine learning algorithms in COVID-19 times," *Land*, vol. 11, no. 11, p. 2100, Nov. 2022. doi:10.3390/land11112100
- [4] J. D. Hulchanski, "The concept of housing affordability: Six contemporary uses of the housing expenditure-to-income ratio," *Housing Studies*, vol. 10, no. 4, pp. 471–491, Oct. 1995. doi:10.1080/02673039508720833
- [5] R. Molloy, C. G. Nathanson, and A. Paciorek, "Housing Supply and affordability: Evidence from rents, housing consumption and household location," *Finance and Economics Discussion Series*, vol. 2020, no. 044, Jun. 2020. doi:10.17016/feds.2020.044
- [6] M.-J. Jun, "The effects of housing preference for an apartment on residential location choice in Seoul: A random bidding land use simulation approach," *Land Use Policy*, vol. 35, pp. 395–405, Nov. 2013. doi:10.1016/j.landusepol.2013.06.011
- [7] X. Shi and Y. Jiang, "Research on house rental recommendation algorithm based on Deep Learning," *Proceedings of the 2022 3rd International Conference on Big Data Economy and Information Management (BDEIM 2022)*, pp. 604–613, 2023. doi:10.2991/978-94-6463-124-1_70

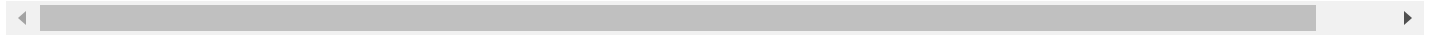
Gantt Chart:

https://docs.google.com/spreadsheets/d/1Xb02M_AjrFq09L2dNq12B_6y__a3cYjiLMz3rHDt0kw/edit?usp=



Contribution Table:

https://docs.google.com/document/d/1hkcrZ9M7G-edv9ugA7WUhuNaagwA5qBMyM-2Exv6_ik/edit?usp=share



Final Video Presentation:

<https://www.youtube.com/watch?v=W3sqRq5UCXY>

Explanations of Directories and Files:

Directory/File	Description
/ML4641_Team1_Midterm.pdf	Midterm report
/README.md	Final report and directory
/Clustering/	Directory for all clustering code and datasets
/Clustering/FINAL_clustering_SL.ipynb	Clustering code
/Clustering/clustering_AP.ipynb	Outdated clustering code, Arian branch
/Clustering/clustering_TA.ipynb	Outdated clustering code, Taiki branch
/Clustering/housing_prices_encoded.csv	Cluster housing prices dataset postprocessed
/Clustering/kaggle_housing_prices.csv	Clustering housing prices dataset
/Clustering/kaggle_realtor_data.csv	Clustering realtor information dataset
/Datasets/	Directory for all datasets
/Datasets/kaggle-breakeven.csv	A dataset for the decision tree
/Datasets/kaggle-housing-prices.csv	Housing prices dataset
/Datasets/kaggle-realtor-data.csv	Realtor information dataset
/Datasets/kaggle-rental-properties/	Rental information dataset

Directory/File	Description
/Datasets/kaggle-rental-properties/event_types.xlsx	An xlsx file for event types
/Datasets/kaggle-rental-properties/kaggle_rental_properties.db	A db file for rental properties
/Datasets/kaggle-rental-properties/kaggle_rental_properties.sqbpro	An sqbpro file for rental properties
/Datasets/kaggle-rental-properties/property.csv	A csv file for properties
/Datasets/kaggle-rental-properties/property.xlsx	An xlsx file for properties
/Datasets/kaggle-rental-properties/property_and_user_activity.csv	A csv file for property and user activity
/Datasets/kaggle-rental-properties/property_and_user_activity_encoded.csv	A csv file for encoded property and user activity
/Datasets/kaggle-rental-properties/user_activity.csv	A csv file for user activity
/Datasets/kaggle-rental-properties/user_activity.xlsx	An xlsx file for user activity
/Datasets/kaggle-rental-properties/Visualizations/	This is the visualization using boxplot an distrogram for deposit and monthly rent
/Datasets/kaggle-rental-properties/Visualizations/deposit_boxplot.png	This is the boxplot of deposit from kaggle-rental-properties
/Datasets/kaggle-rental-properties/Visualizations/deposit_histogram.png	This is the histogram of deposit from kaggle-rental-properties
/Datasets/kaggle-rental-properties/Visualizations/monthly_rent_boxplot.png	This is the boxplot of monthly rent from kaggle-rental-properties
/Datasets/kaggle-rental-properties/Visualizations/monthly_rent_histogram.png	This is the histogram of monthly rent from kaggle-rental-properties
/Decision Tree/	Directory for all decision tree code

Directory/File	Description
/Decision Tree/FINAL_rent_vs_buy_AP_FF1.ipynb	Decision tree code
/Decision Tree/rent_vs_buy_AP.ipynb	Outdated decision tree code, Arian branch
/Decision Tree/rent_vs_buy_SL.ipynb	Outdated decision tree code, Sean branch
/Model Visualizations/	Directory for all figures used in report
/Model Visualizations/Clustering Visualizations/	Directory for all clustering figures used in report
/Model Visualizations/Clustering Visualizations/dbscan.png	DBSCAN clustering plot
/Model Visualizations/Clustering Visualizations/gmm.png	GMM clustering plot with 6 clusters
/Model Visualizations/Clustering Visualizations/gmm_bic.png	Plot Bayes Information Criterion versus number of clusters
/Model Visualizations/Clustering Visualizations/housing_prices_area.png	Box plot of housing price depending on the area before cleaning data
/Model Visualizations/Clustering Visualizations/housing_prices_baths.png	Box plot of housing price depending on the number of baths before cleaning data
/Model Visualizations/Clustering Visualizations/housing_prices_beds.png	Box plot of housing price depending on the number of beds before cleaning data
/Model Visualizations/Clustering Visualizations/housing_prices_price.png	Box plot of housing price depending on the price before cleaning data
/Model Visualizations/Clustering Visualizations/housing_prices_stories.png	Box plot of housing price depending on the number of stories before cleaning data
/Model Visualizations/Clustering	K-Means clustering plot with

Directory/File	Description
Visualizations/kmeans_2.png	2 clusters
/Model Visualizations/Clustering Visualizations/kmeans_3.png	K-Means clustering plot with 3 clusters
/Model Visualizations/Clustering Visualizations/kmeans_4.png	K-Means clustering plot with 4 clusters
/Model Visualizations/Clustering Visualizations/kmeans_elbow.png	A plot displaying how well a data set was clustered using elbow method
/Model Visualizations/Decision Tree Visualizations/	Directory for all decision tree figures used in report
/Model Visualizations/Decision Tree Visualizations/DT_cleanData_v1.png	Box plots of deposits before and after data cleaning
/Model Visualizations/Decision Tree Visualizations/decision_tree_visualization.svg	Visualization of the decision tree
/Model Visualizations/Decision Tree Visualizations/dt_0_viz.svg	Old visualization of the decision tree
/Model Visualizations/Decision Tree Visualizations/dt_1_viz.svg	Old visualization of the decision tree
/Model Visualizations/Decision Tree Visualizations/dt_2_viz.svg	Old visualization of the decision tree
/Model Visualizations/Decision Tree Visualizations/recursive_feature_elimination.png	Recursive feature elimination with correlated features
/Model Visualizations/Regression Visualizations/	Directory for all regression figures used in report
/Model Visualizations/Regression Visualizations/regression_lasso.png	Lasso regression plot actual vs predicted
/Model Visualizations/Regression Visualizations/regression_linear.png	Linear regression plot actual vs predicted
/Model Visualizations/Regression Visualizations/regression_logistic.png	Logistic regression plot actual vs predicted

Directory/File	Description
/Regression/	Directory for all regression code
/Regression/FINAL_regression.ipynb	Regression code