Rentalytics: Analyzing Trends in Apartment Rent Pricing

By: Ankit Singh Chauhan, Sai Aravind Donga, Kael Ecord, and Hari Shivani Gudi

Abstract

Estimating the cost of renting an apartment is a big task with real-world applications in many different industries. Proper forecasting may help landlords establish competitive pricing, maximize investment profits, and guarantee high occupancy rates. From the tenant's perspective, accurate forecasts facilitate well-informed decision-making, allowing for comparison with real listings, identifying properties that may be overpriced or undervalued, and negotiating better conditions to save money. In the real estate sector, effective flat rental price forecasting has the potential to enhance decision-making, resource allocation, and quality of life for both renters and landlords.

Introduction

Literature Review

Several research have investigated predicting apartment rental costs with a variety of methods and information sources. To forecast rental costs in New York City, Jiang et al. (2019) combined machine learning algorithms with conventional statistical techniques. They discovered that include neighbourhood-level variables, such crime rates and accessibility to facilities, increased the precision of their forecasts. Choy and Ho (2023), in their work, demonstrated the effectiveness of machine learning algorithms like Extra Trees (ET), k-Nearest Neighbors (KNN), and Random Forest (RF) in predicting property prices compared to traditional hedonic price models. Their study, using a dataset of 24,936 housing transaction records, shows that these algorithms outperform traditional statistical approaches, demonstrating the potential of machine learning to improve price forecast accuracy in the real estate market. Another study by Nguyen et al. (2020) utilized web-scraped data from online rental listings to predict prices in major U.S. cities. They employed natural language processing (NLP) techniques to extract relevant features from textual descriptions of rental listings. These extracted textual features were combined with other structured data to build predictive models, which enhanced the predictive performance compared to traditional models. Finally, Chaplin et al. (2021) developed a rental price prediction model considering socioeconomic characteristics, housing market conditions, and neighbourhood amenities. Their work aimed to provide insights for policymakers and urban planners to address issues related to affordable housing and gentrification by incorporating factors like socioeconomic characteristics, housing market conditions, and neighbourhood amenities.

Even though these studies have greatly advanced the field, there is still more work to be done to reliably anticipate flat rental costs, especially considering the development of new data sources and sophisticated methodologies. To address this issue, the attached document looks to investigate a few regression and classification models that incorporate a variety of pertinent characteristics.

Motivation

Accurately predicting apartment rental prices is crucial for both landlords and tenants. Landlords can maximize investment returns and ensure high occupancy rates, while tenants need affordable housing options. Accurate predictions offer valuable insights to stakeholders in the real estate industry, enabling data-driven pricing strategies and guiding clients towards suitable properties. For tenants, accurate predictions enable informed decisions, comparison with actual listings, identification of potential over- or underpriced properties, and negotiation of better terms, leading to cost savings. Overall, efficient prediction of apartment rental prices has significant implications across various sectors, improving decision-making, resource allocation, and quality of life for both landlords and tenants.

Methodology

Dataset Background

The dataset that was used for this project is titled "Apartment for Rent Classified" and can be found online at the UC Irvine Machine learning Repository. (Apartment for Rent Classified - UCI Machine Learning Repository). The dataset contains 99,492 rows of 22 columns, where each row represents information about 1 apartment listing.

Data Exploration, Preprocessing, and Cleaning

Cleaning

Because this dataset has so many rows, we want to ensure that we are looking at observations that all fit some base criteria. For the scope of this analysis this we will include all listings in the data that are:

1. Apartments (determined by the category column)
2. Rent is paid monthly (price_type = 'Monthly')
3. State column has value (!is.na(state))

Observations that do not meet this criterion will be deleted from the dataset as a different model would be best in determining their price.

As mentioned earlier, the dataset contains 22 columns. However, some of these columns are not particularly useful for the task at hand and have therefore been removed. The following columns have been removed from the dataset:

- Id – Listing ID
- Category – Describes type of listing. Removed after performing the above filtering.
- Title – Apartment listing title
- Body – Lengthy textual description of apartment listing. NLP is required to implement (outside the scope of this class).
- Amenities – Variable length text column listing the amenities at a given apartment. Hard to use as some listing may classify what an amenity is. For example, is Washer/Dryer an amenity or expected? Due to this problem, we will simply remove this column from the data.

- Pets_allowed – Variable length string variable designating whether cats and or dogs are allowed.
    - See the ***Preprocessing*** section to see how this was implemented.
- Currency – Currency of monthly payment (all USD).cret
- Price_display – Apartment listing price as string for displaying.
    - Example value: $750
- Price_type – How often rent is paid. (Data filtered to only include “Monthly”)
- Address – Listing address.
- Cityname – US city listing in located in.
    - Too many US cities to use as categorical variable.
- Source – Website the listing originated from.
- Time – When classified listing was created. Not in standard date/time format.
    - Example value:1568753820

Preprocessing

The raw dataset includes a few columns that contain information that might be useful when predicting the monthly rent of an apartment, however they are either not in the correct format or in a more complex format than we want to deal with. One example of this is the title column. This column contains the title of the apartment listing. Now there are some more involved NLP techniques that could be useful, however we will stick with a more novel approach. This column was turned into a binary column titled “studio”. If the title of the listing contains the word “studio” then we placed a 1 in the studio column and a 0 otherwise. A similar process of converting a string-based column to a binary column was also performed on the “pets_allowed” column. This was split into two columns, cats_allowed and dogs_allowed, where 1 represents yes and 0 represents no. The final step of preprocessing the data was to create a column for the division and region of the country an apartment listing was in. The regions and divisions were determined by the US Census Bureau. The categorical target variable, "price_category," was derived by classifying the continuous variable "price" into three unique classes—"Low," "Medium," and "High"—based on the 33rd and 67th percentiles, respectively. The dataset was then divided into training and test sets using a random 80/20 split. For classification, the columns "state," "us_division," and "us_region" were removed from the dataset, leaving only numerical columns for analysis.

Exploration

After all cleaning and preprocessing the final dataset that will be used for exploration and model building contains 99,125 observations and 14 columns for each observation. That leaves 13 columns for input and the price (monthly rent) as the response variable.

All 50 states, plus DC, are represented in the dataset with Texas and California being the states having the most listings, 11,255 and 10,308 respectively. It should be noted that there are a handful of states with extremely low representation in the dataset. This with less than 100 observations are: Arkansas, DC, Delaware, Hawaii, Idaho, Maine, Montana, New Mexico, South Dakota, West Virginia, and Wyoming. This fact was the primary reason for creating other forms of grouping US areas together like region and division.

When looking specifically into the response variable price, we found that the distribution was slightly skewed by some very large values. The mean price was $1,525, while the median

was $1,350. For now, the extremely large values (max $52,500) for price will be left in, however when building the models in the future we will look at Cook's D and other values to determine if these points are having too much influence on the models. If so, they will be removed from the dataset, which will further constrain the scope of our models.

Variable Selection and Model Comparison

The best subset method will be used to select the variables to include in all multiple regression models and other regression-based models. Upon performing this method BIC, Adjusted-R^2, and Mallow's Cp will be used to compare models. All models created will be tested on the same test set, which will be comprised of a random sample of 20% of the full data. The test means squared error (MSE) will be used to compare all regression models and accuracy will be used to compare classification models.

Results

Regression

For classification the dataset was preprocessed by loading it and following the above preprocessing steps. As mentioned in the *methods* section all regression models will be compared via their test MSE.

Model 1 – Simple Linear Regression

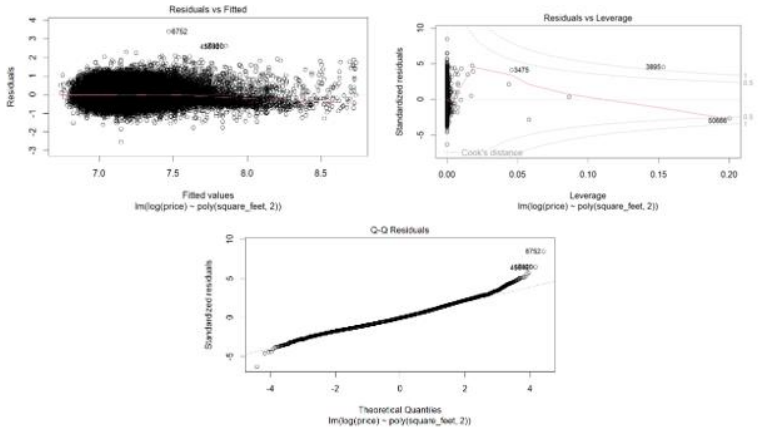
For this first model we will set a baseline for future and more advanced models to beat. The most important variable when predicting price is square feet. For this reason, we will start with a simple linear regression model that uses only square feet to predict price. When looking at the response variable price, we saw that it was slightly skewed to the right. Because of this we for the initial SLR model we will log transform the response variable to reduce the skewness. The first model will take the following form:

$$\log(price) \ = \ \beta_0 + \beta_1(squarefeet) + \beta_2(squarefeet^2) \ + \varepsilon$$

While all variables included in the model are very significant, looking at the diagnostic plots provides a different story. We can see below in Figure 1 that the residuals appear to not be normal, a few points have significantly higher leverage than most observations, and that there are some common points causing problems on all three of these diagnostic plots.

Figure 1

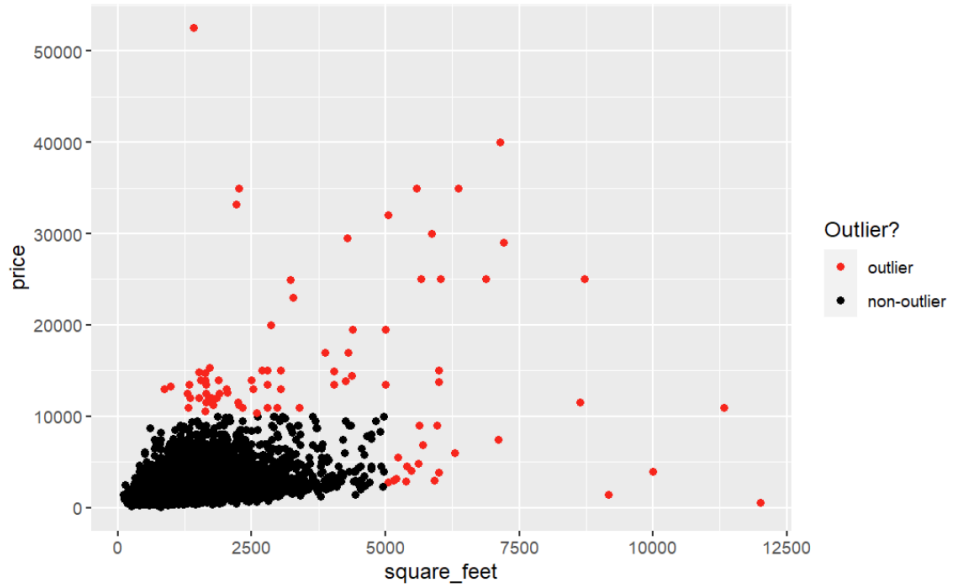
SLR Diagnostic Plots



Diving deeper into these we can see a simple scatter plot plotting price against square feet in Figure 2 shows that outside of a range of values we have very few observations. The red dots represent points that have a monthly rent higher than $10,000 or apartments that are larger than 5,000 square feet. Out of over 95,000 observations there are only 89 observations that fit this criterion. Having these few observations with such high values further justifies the next step of removing these observations from the data set.

Figure 2

Finding outliers with price vs square feet



NOTE: Observations in red will be removed from dataset

After removing these points from the data, we will use the following model:

$$\log(price) = \beta_0 + \beta_1(squarefeet) + \varepsilon$$

When using this model, we find that all variables in the model are once again significant. However, the test MSE is quite high, see Table 3. This shows that the simple linear regression approach isn't telling the full story and that incorporating other variables by introducing a more complex model will likely produce better results.

Model 2 – Multi Linear Regression

The second regression model we use to predict price is a multiple linear regression model. The first step when creating this model is to ensure that we remove correlated variables. This will be done by creating a model will all variables and computing the VIF values for each variable. If a variable has a VIF value greater than 3 it will be removed from the pool of variables. The result of this process removes the variables, state, us_division, longitude, latitude, and cats_allowed from the pool. The final VIF values for the variables left in the pool can be found in Table 1.

Table 1

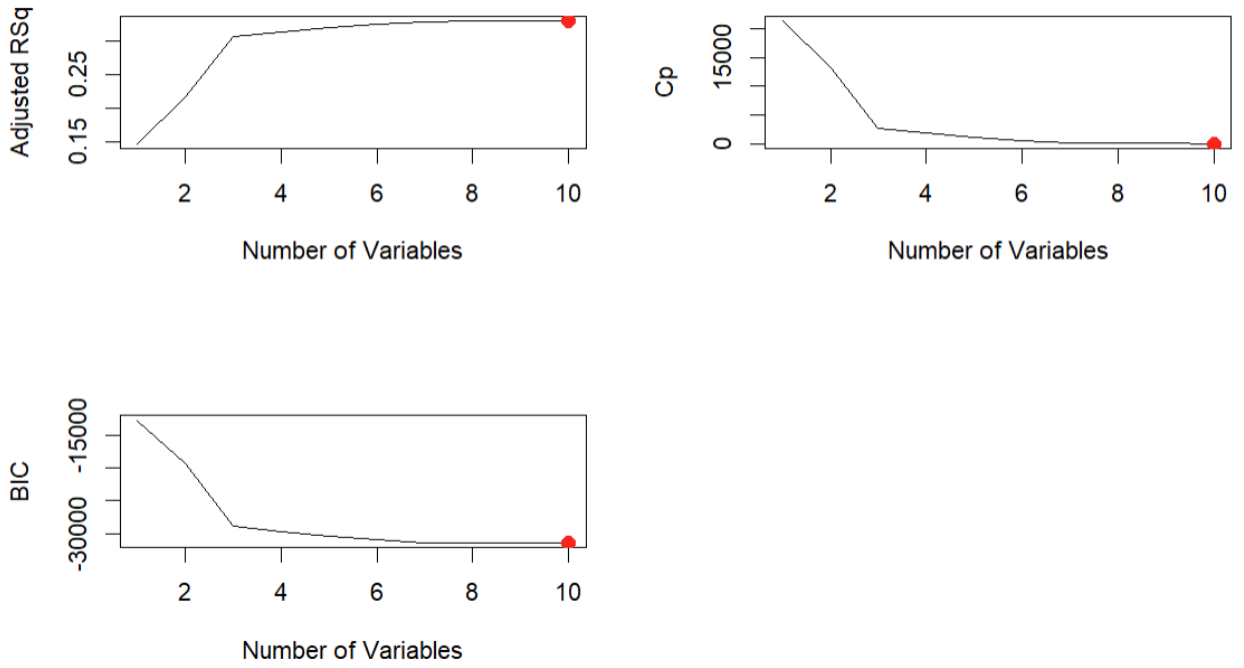
Final Variable VIF values

Variable	DF	GVIF	GVIF^(1/(2*DF)
bathrooms	1	2.402524	1.550008
bedrooms	1	2.387365	1.545110
fee	1	1.004115	1.002055
has_photo	1	1.006939	1.003463
square_feet	1	2.741262	1.655676
studio	1	1.069346	1.034092
dogs_allowed	1	1.011284	1.005626
us_region	3	1.043224	1.007078

With the variables listed in Table 1 we used the best subset selection method to produce the best model for each subset size from 1 to 10. Using the regsubssets function, we found that the most important variable was once again square feet. This was shown by being the best single variable further confirming the statement made when discussing the SLR model. To select the best subset size, we will look at the graphs in Figure 3. These show the values for adjusted R-squared, Mallow's Cp, and BIC values for the best subset for each size.

Figure 3

Best Subset Results



In Figure 3 when can see that when the number of variables is 10, we have the lowest value for all 3 metrics. However, if we look slightly to the left, we can see that after we hit 7 variables there is little to no change. For this reason, we will use the 7 variable model when discussing the chosen best subset model. The variables included in this model are bathrooms, bedrooms, square_feet, studio, and all levels of us_region. Upon building the model we can see that all

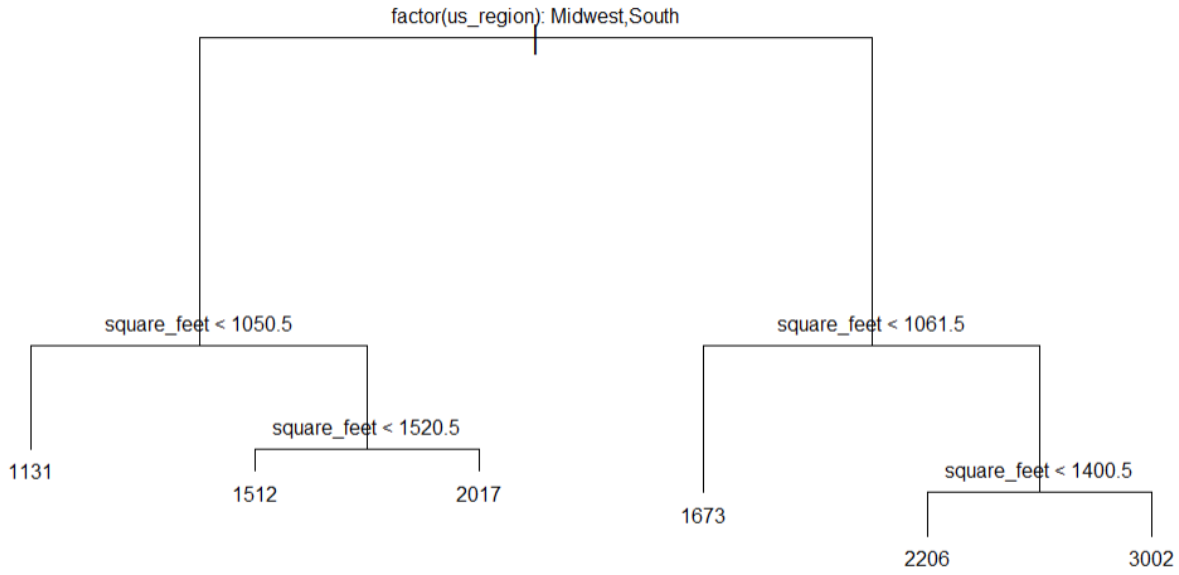
variables are extremely significant, have VIF values lower than 3, and that the test MSE is much improved over the SLR model, which we expected.

Model 3 – Regression Trees

The next model used to predict price was a regression tree. Variables us_division, state, longitude, and latitude were removed from the variables in the model due to singularity issues when including them all. This is because knowing one value can tell you the value of another. For example, if you have a state value then you also know the values for us_region and division. Building the tree produces an interesting result, which can be seen in Figure 4 below.

Figure 4

Regression Tree Result



The top split represents the variable that provided the most distinction between the chosen classes when predicting price. Here we can see that the Midwest and South regions had similar values while the Northeast and West had similar values. All splits made further down the tree were based on the square footage of the apartment. This tells us two important things. On average apartments in the Midwest and South have a lower cost per square foot, where more expensive apartments are generally seen out West and in the Northeast. This finding goes along with general intuition. When using this model to predict values for the test set, we got a test MSE that was slightly higher than the MLR model, but lower than the test MSE seen in the SLR model, which can be seen in Table 3.

Model 4 – Ridge Regression

The final regression model implemented to predict price was a ridge regression model. For this model all variables were included. Once the base model was trained using the base glmnet function, we used cv.glmnet to select the best lambda value. We found that 29.857 was the best value. Using this as the value for lambda we retrained the model on the train set and tested the model using the test set. The final test MSE can be found in Table 4.

Classification

For classification the dataset was preprocessed by loading it and following the above preprocessing steps. This included removing any rows with missing values using the na.omit function. As mentioned in the *methods* section all classification models will be compared via their testing accuracy.

Model 1- Multinomial Logistic Regression

The logistic regression model was trained on the training set of data using the *multinom* function from the *nnet* package. All available features were regressed against the target variable "price_category". Using the *predict* function, the trained model generated predictions for the test dataset. To determine the accuracy of the logistic regression model, these predictions were compared with the actual values found in the test data.

Model 2-QDA

The *qda* function from the MASS package was used to build the quadratic discriminant analysis (QDA) model. Using the target variable "price_category" and all available features as predictors, this model was trained on the training dataset. Using the *predict* function, the QDA model was used to predict the classes in the test data after training. The QDA model's accuracy was then determined by comparing the actual values found in the test data with the predicted classes.

Model 3-LDA

The "MASS" package has been imported and the LDA model was trained on the training dataset. The model is constructed with the target variable "price_category" and all available features serving as predictors. Following the training phase, the trained LDA model was used to predict the classes in the data tested using the *predict* function. The accuracy of the LDA model was then calculated.

Model 4-KNN

To implement the K-Nearest Neighbors (KNN) technique, a loop was created to run across a range of k values (from 1 to 10). For each value of k, the KNN model was trained on the training data using the *knn* function from the "class" package. The accuracy of the KNN model was calculated for each value of k by comparing the predicted classes to the actual classes observed in the test data. Finally, the best value of k, which corresponds to the maximum accuracy attained, has been determined.

Compiled results

Table 3

Regression Model Test MSE Results

Model	Test MSE
Simple Linear Regression	504417.37
Multiple Linear Regression	376273.01
Regression Tree	402919.52
Ridge Regression	**279035.36**

Table 4

Classification Model Test Accuracy Results

Model	Test Accuracy
Multinomial Logistic Regression	**0.9998484**
QDA	0.9147001
LDA	0.7682551
KNN (K=1)	**0.9975744**

Table 5

Test accuracy for k = 1 through k =10

K value	Test Accuracy
1	**0.9975744**
2	0.9969175
3	0.9968669
4	0.9965132
5	0.9966143
6	0.9969175
7	0.9972712
8	0.9971701
9	0.9972712
10	0.9972207

Figure 5

Confusion matrices for Classification models

```
> print(conf_matrix)
             multi_logistic_pred
actual_values  Low Medium High
       Low    6409      0    0
       Medium    3   6812    0
       High      0      0 6565
> print(conf_matrix_knn)
             knn_pred
actual_values  Low Medium High
       Low    6397     12    0
       Medium   23   6786    6
       High      1      6 6558
```

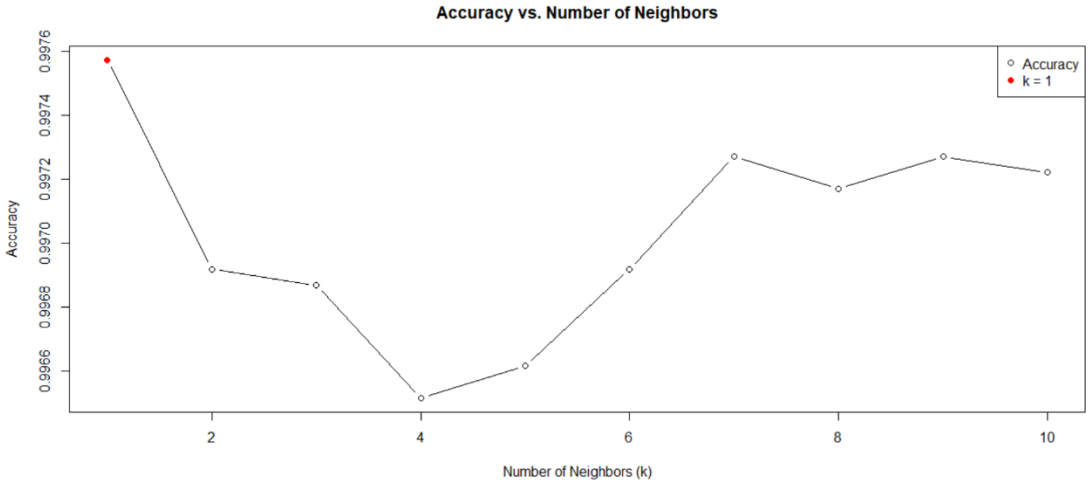
```
> print(confusion_matrix_lda)

actual_values  Low Medium High
       Low    5101   1308    0
       Medium 1362   5391   62
       High      8   1846 4711
> print(confusion_matrix_qda)

actual_values  Low Medium High
       Low    5884    367  158
       Medium   95   6577  143
       High      0    925 5640
```

Figure 6

Performance of a K-Nearest Neighbors model



Discussion

Regression

Performance

When comparing various regression techniques, we saw that the ridge regression method had the best mean-squared error on the test set. For this reason, we chose this as the best model. The use of MLR, SLR, and regression trees showed us the most important variables to consider when predicting apartment prices. We consistently saw that square_feet performed the best as the individual predictor for the SLR and MLR models while also playing a key role in the tree-based method. Other variables that contributed heavily were us_region, bedrooms, and bathrooms. These results make sense and validate prior assumptions and research. Overall, there is a lot of room for improvement from our models. While ridge regression performed better than the other models, we tested, the value of the test MSE is quite high. Our research results show that to be super accurate, we need more information to inform the models.

Limitations for regression -

Even though we had a lot of observations, over 90,000, there were regions for key variables such as price and square_feet that did not have much representation. For this reason, we had to restrict our analysis to only include observations where the monthly rent was less than $10,000/month or were more than 5,000 square feet. The reason for this was because there were so few points in this region that these points had a significantly higher impact on the model than other points and we wanted to make sure that wasn't the case. Another limitation that we found was that the data had a lot of information stored in long strings. Being able to apply NLP techniques to extract key information from these strings would most likely prove very useful in improving the accuracy of the models. An example of how this would prove to be useful was in the *body* column, which was deleted. This column contained the full description of the listing, which sometime would contain words like luxurious or spacious. NLP techniques would allow the model to use this information to inform its predictions.

Classification

Performance

Our study compared various models for classifying apartment rental prices. Multinomial Logistic Regression achieved an accuracy of 99.98%, but this model assumes linearity in log odds, which may not always be true in complex datasets. Quadratic Discriminant Analysis (QDA) achieved an accuracy of 91.47%, demonstrating its ability to handle non-linear relationships between predictors and classes. Linear Discriminant Analysis (LDA) produced an accuracy of 76.83%, which is lower than both logistic regression and QDA. LDA generally assumes a multivariate normal distribution within each class. K-Nearest Neighbors (KNN) achieved high accuracies ranging from 99.65% to 99.73% across different values of k, with the highest accuracy observed at k=1. KNN is a non-parametric method that makes minimal assumptions about the underlying data distribution, making it flexible and robust. The graph in *Figure 6* illustrates the performance of a KNN model as the number of neighbors considered increases. The model's accuracy decreases with fewer neighbors, indicating insufficient information for accurate predictions. As the number of neighbors increases from 4 to 7, the accuracy improves, indicating better data capture and predictions. After k=7, adding more neighbors doesn't significantly improve accuracy, and may even lead to a slight decrease.

Limitations for classification models:

- The Multinomial Logistic Regression model showed high AIC and residual values
- Higher Test Accuracy for KNN (K=1) Compared to Other K Values
- Lack of proper feature selection

These points could lead to poor predictive performance and excessive sensitivity to noise in the training data.

Future Work

It's essential to refine the model by potentially incorporating additional relevant predictors to improve model fit and predictive accuracy. To mitigate overfitting, the model should be evaluated on validation data or cross-validation, and tuning the value of k or exploring alternative classification algorithms may help improve the model's generalizability.

References

Chaplin, S., Jones, L., & Edwards, P. (2021). Developing a rental price prediction model to assist policymakers. Housing Policy Journal, 29(4), 410-428.

Choy, L. H., & Ho, W. K. (2023, March 25). *The Use of Machine Learning in Real Estate Research*. Land. https://doi.org/10.3390/land12040740

Nguyen, H., Smith, J., & Doe, A. (2020). Utilizing web-scraped data for rental price predictions in major U.S. cities. Journal of Urban Economics and Management, 34(2), 120-135.

Pai, P.-F., & Wang, W.-C. Using machine learning models and actual transaction data for predicting real estate prices. Applied Sciences-Basel, 2020, 10 (17), 5832. 10.3390/app10175832.

Wikipedia contributors. (2024, April 10). List of regions of the United States. Wikipedia. https://en.wikipedia.org/wiki/List_of_regions_of_the_United_States