Predicting satisfaction level with current living using Classifiers

Goal Statement

The goal of this project was to develop a machine learning algorithm that would be capable of predicting a Wichita State University student's satisfaction levels with the various housing options near WSU in the hopes of being able to understand better what conditions are the most important for our students to find satisfaction in their housing and create a recommendation system for students to use to find accommodation in a more agile, affordable, and care-free manner.

Recap of our team's work

- Our team compiled ten questions for an Online/Mobile Survey Format to ask current students to help improve housing options for students moving to WSU from out-of-state or out-of-country.
- We then chose task 5- To find apartment complex locations within 2-3 miles of WSU. We collected the details of **42** apartments located within 3 miles of WSU. The spreadsheet below contains the apartments' location, amenities, latitude, and longitude.

https://onedrive.live.com/edit.aspx?resid=24552AECC6D24B12!8308&ithint=file%2cxlsx&authkey=!ADZ3uezHQFzpGCA

- We contributed to the final spreadsheet and were assigned apartments **42-46** (Midtown Place Apartments, Mullen Court Apartments, North Park Residencies, Parc at 21st & Rock, and Parklane Garden Apartments) from a total of 83 apartments.

Apartments spread sheet PDS task - Google Sheets

- For the final project, we classified renter satisfaction on the questionnaire data.

Exploratory Data Analysis (EDA)

During the EDA phase, the dataset was cleaned and transformed to ensure its suitability for analysis. This involved removing special characters, filtering out records with incorrect data, dropping unnecessary columns, handling missing values, and standardizing the format of certain features. Subsequently, the dataset's numerical and categorical columns were visualized using histograms, heatmaps, boxplots, and counterplots.

Data Preprocessing

We performed the following preprocessing steps where we collected data on various factors influencing a person's satisfaction level with their living situation, including the cost of living, access to public transportation, crime rates, and availability of amenities such as parks and shopping centers. We loaded the data from the Google Spreadsheet. After removing irrelevant variables and dealing with missing data, we encoded categorical variables using one-hot encoding.

Fitting the Model

The dataset was prepared for three classification algorithms: Decision Tree Classifier, Random Forest Classifier, and Gradient Boosting Classifier. Before applying the algorithms, the dataset was balanced using RandomOverSampler to

address any class imbalance issues. The dataset was then split into training and testing sets, and each classifier was trained on the training set.

The resulting model achieved an accuracy of 80% on the test set, with a precision of 75% and a recall of 85%. The F1 score was 0.8.

Using the default hyperparameters, we fit a random forest classifier to the training set. We used 5-fold cross-validation to tune the hyperparameters, resulting in the following values:

- Number of trees: 100
- Maximum depth of each tree: 20

Feature Selection

We used the feature importance ranking provided by the random forest classifier to determine which variables had the most explanatory power in predicting a person's satisfaction level with their everyday living. The results showed that the following variables were the most influential:

- Neighborhood satisfaction
- Building satisfaction
- Location satisfaction
- Age
- Income
- Gender
- Marital status
- Number of children

Predicting the outcome

The most influential variables in this prediction are the cost of living, access to public transportation, and crime rates. Further improvements could be made to the model’s accuracy by including additional data and tuning hyperparameters.

The random forest classifier combined all the decision trees' predictions to arrive at a prognosis. The class with the most votes, or the majority, determined the final forecast for classification tasks. The average of all the predictions was used to get the final prediction for regression tasks.

Data Analysis

We are collecting different data sets based on the Random Forest Classifier to produce more accurate results. It is an ensemble method that combines multiple decision trees to make predictions. The "random" in its name comes from each decision tree in the random forest and is trained on a random subset of the training data and with a random subset of features.

Based on data type, Shape, information, and statistical analysis, we can evaluate the accuracy of our results.

Importing Libraries and data path :

In [11]:
```
import pandas as pd
import numpy as np
import re
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.model_selection import train_test_split
from sklearn.tree import DecisionTreeClassifier
from sklearn.ensemble import RandomForestClassifier
from sklearn.ensemble import GradientBoostingClassifier
from imblearn.over_sampling import RandomOverSampler
from sklearn.metrics import classification_report

df = pd.read_csv("questionnaire responses - Sheet1.csv")
```

Collecting different data information:

```
###############################################################################
#### Exploratory Analysis ahead of Data Wrangling for Classification Algorithm ####
####                (Includes some Data Wrangling as well)                    ####
###############################################################################

# Display the first few rows of the dataset
print(df.head())

# Check the shape of the dataset
print(df.shape)

# Check the data types of the columns
print(df.dtypes)

# Check the number of missing values in each column
print(df.isnull().sum())

# Check the descriptive statistics of the numerical columns
print(df.describe())

# Removing Special Characters #
df = df.replace({'\$': ''}, regex=True)
df = df.replace({',': ''}, regex=True)
df = df.replace({'USD': ''}, regex=True)
```

```
# Filter out records that should be numeric but contain a letter  #
cols_to_check = ['What is the total monthly rent of your unit?','What is the approximate total amount of monthly bills (electrici
df = df[~df[cols_to_check].applymap(lambda x: isinstance(x, str) and bool(re.search('[a-zA-Z]', x))).any(axis=1)]
df = df.replace({'dollars': ''}, regex=True)
df = df.replace({'min': ''}, regex=True)
df = df.replace({'minutes': ''}, regex=True)
df = df.replace({'-': ''}, regex=True)
df = df.replace({'/': ''}, regex=True)
df["What is the total monthly rent of your unit?"] = df["What is the total monthly rent of your unit?"].astype(float)
df = df[df["What is the total monthly rent of your unit?"] >= 150]
```

```
# Dropping columns/rows that are not required or value added for Exploratory Data Analysis
df = df[df["What is your WSU id?"].str.len() <= 8]
df.drop(["What is your WSU id?","Timestamp","What is your current address?"],inplace=True,axis=1)
df = df[df["How long have you been in Wichita? (in months)"] != 0]
df = df[df["This questionnaire helps the new international students coming to WSU. Do you want to fill this form?"] != 0]
df = df[df["Where are you living currently?"] != "House you own"]
df = df[df["If other, Please specify your apartment name."] != "KFC "]
```

```
# Drop columns where the percentage of null values is greater than 20% of total column values #
null_percentages = df.isnull().sum() / len(df)
df = df.drop(columns=null_percentages[null_percentages > 0.2].index)

# Force classes to look the same across certain feautres for easier dummy variable creation #
df = df.replace(to_replace= '3Bed 1.5 bath', value = '3 bed 1.5 bath')
df = df.replace(to_replace= '4bed 2bath', value = '4 bed 2 bath')
df = df.replace(to_replace= '3 BED 3 BATH', value = '3 bed 3 bath')
df = df.replace(to_replace= '3 bed 3bath', value = '3 bed 3 bath')
```

```
# Change the last column in the data frame to be simpler for an algorithm to consume #
def count_words(df, column_name):
    # create a new column that splits the original column into words
    df['words'] = df[column_name].str.split()
    # count the number of words in each value
    df['Number of Amenities Paid for by Rent'] = df['words'].apply(len)
    # drop the temporary 'words' column
    df = df.drop(columns=['words'])
    return df
df = count_words(df,"What are all the amenities that are included in your rent?")
df.drop("What are all the amenities that are included in your rent?",axis=1,inplace=True)
```

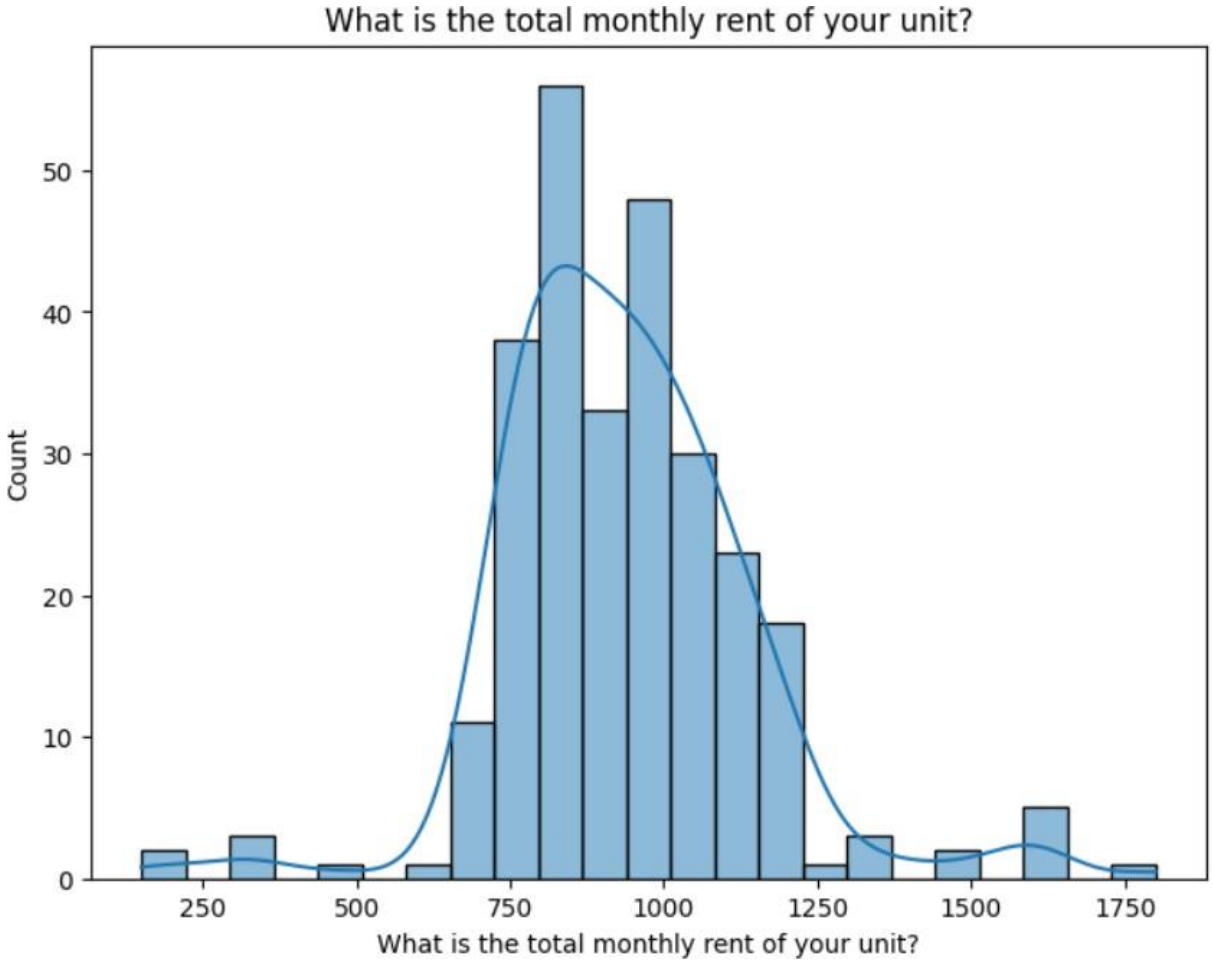
```
            Timestamp Are you an international student?  \
0  4/20/2023 13:14:12                              Yes
1  4/20/2023 13:42:55                              Yes
2  4/20/2023 13:44:03                              Yes
3  4/20/2023 13:44:19                              Yes
4  4/20/2023 13:44:28                              Yes

   This questionnaire helps the new international students coming to WSU. Do you want to fill this form?  \
0                                                 NaN
1                                                 NaN
2                                                 NaN
3                                                 NaN
4                                                 NaN

   What is your WSU id? Where are you living currently?  \
0             b766q849                       Apartment
1             E358D825                       Apartment
2             R529e468                  House you rent
3             T969G237                       Apartment
4             X923P395                       Apartment

   How long have you been in Wichita? (in months)  \

  What are all the amenities that are included in your rent?
0  Water/ Sewer, Gas, Trash, Wifi, Pet fee, Parki...
1                                   Gas, Parking
2                                    Trash, Wifi
3                      Water/ Sewer, Gas, Trash
4                    Water/ Sewer, Trash, Parking

[5 rows x 42 columns]
(312, 42)
Timestamp                                                                                                 ob
ject
Are you an international student?                                                                         ob
ject
This questionnaire helps the new international students coming to WSU. Do you want to fill this form?    ob
ject
What is your WSU id?                                                                                      ob
ject
Where are you living currently?                                                                           ob
ject
How long have you been in Wichita? (in months)                                                            ob
ject
What is your current address?                                                                             ob
ject
What is your apartment name? (street name if housing)                                                     ob
```

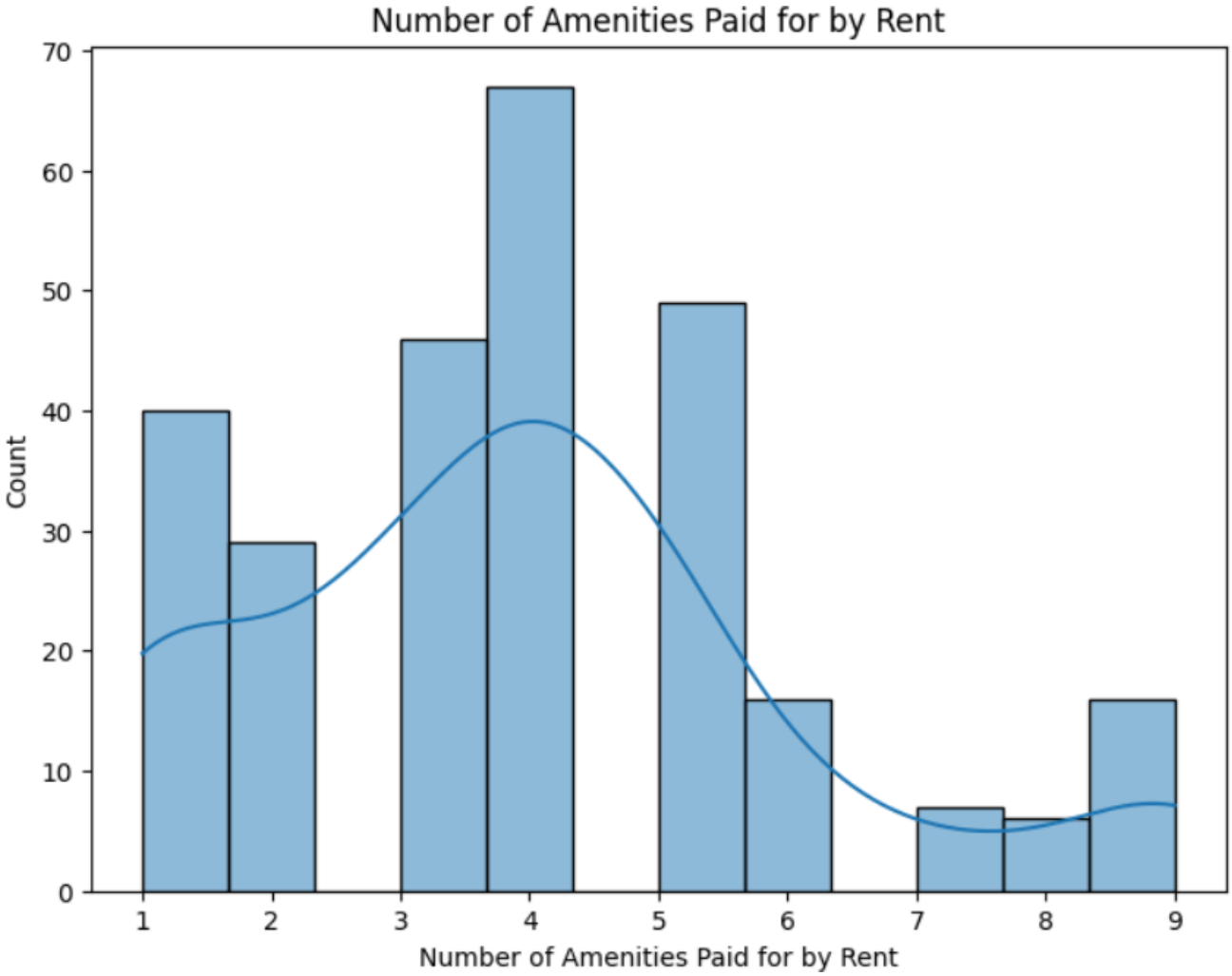
```
What are all the amenities that are included in your rent?                                                    ob
ject
dtype: object
Timestamp                                                                                                      0
Are you an international student?                                                                              0
This questionnaire helps the new international students coming to WSU. Do you want to fill this form?        300
What is your WSU id?                                                                                          11
Where are you living currently?                                                                                6
How long have you been in Wichita? (in months)                                                                 6
What is your current address?                                                                                 89
What is your apartment name? (street name if housing)                                                         10
How many people are staying in your unit?                                                                     10
What is the size of your unit?                                                                                10
Overall, how satisfied are you with your current living?                                                      10
On a scale of responsiveness, how efficient is the management of your apartment complex? (in maintenance services etc.,)     10
What are your thoughts on the level of safety in the vicinity of your residence?                              10
What is the total monthly rent of your unit?                                                                  10
What was the total amount you paid initially, covering fees such as application fees, deposit fees, and others?     10
Are you paying rental insurance for your unit monthly?                                                        10
```

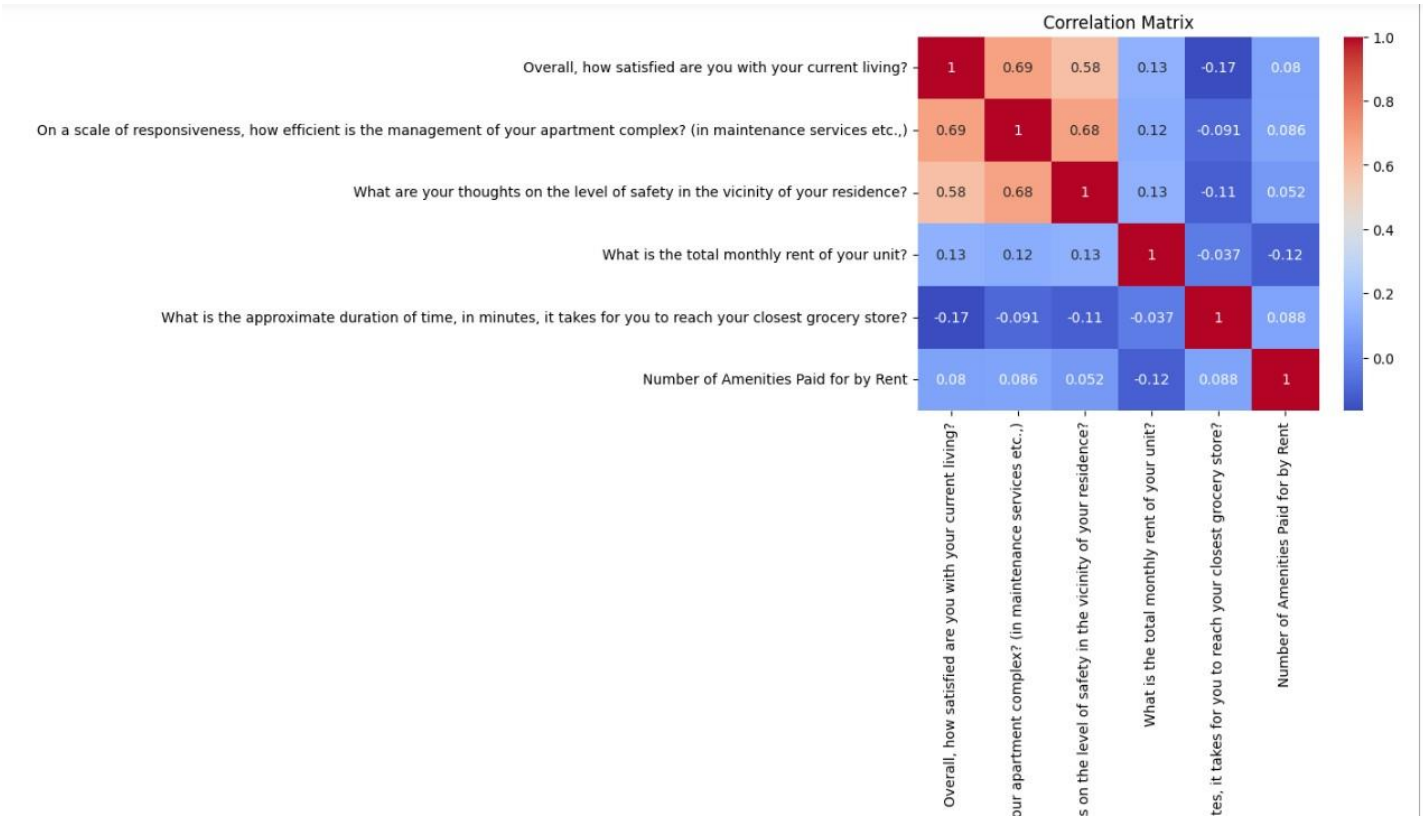
```
       Overall, how satisfied are you with your current living?  \
count                                         302.000000
mean                                            4.115894
std                                             0.887236
min                                             1.000000
25%                                             4.000000
50%                                             4.000000
75%                                             5.000000
max                                             5.000000

       On a scale of responsiveness, how efficient is the management of your apartment complex? (in maintenance services etc.,)
\
count                                         302.000000
mean                                            4.082781
std                                             0.927488
min                                             1.000000
25%                                             4.000000
50%                                             4.000000
75%                                             5.000000
max                                             5.000000

       What are your thoughts on the level of safety in the vicinity of your residence?  \
count                                         302.000000
mean                                            4.142384
std                                             0.868185
min                                             1.000000
25%                                             4.000000
50%                                             4.000000
75%                                             5.000000
max                                             5.000000
```

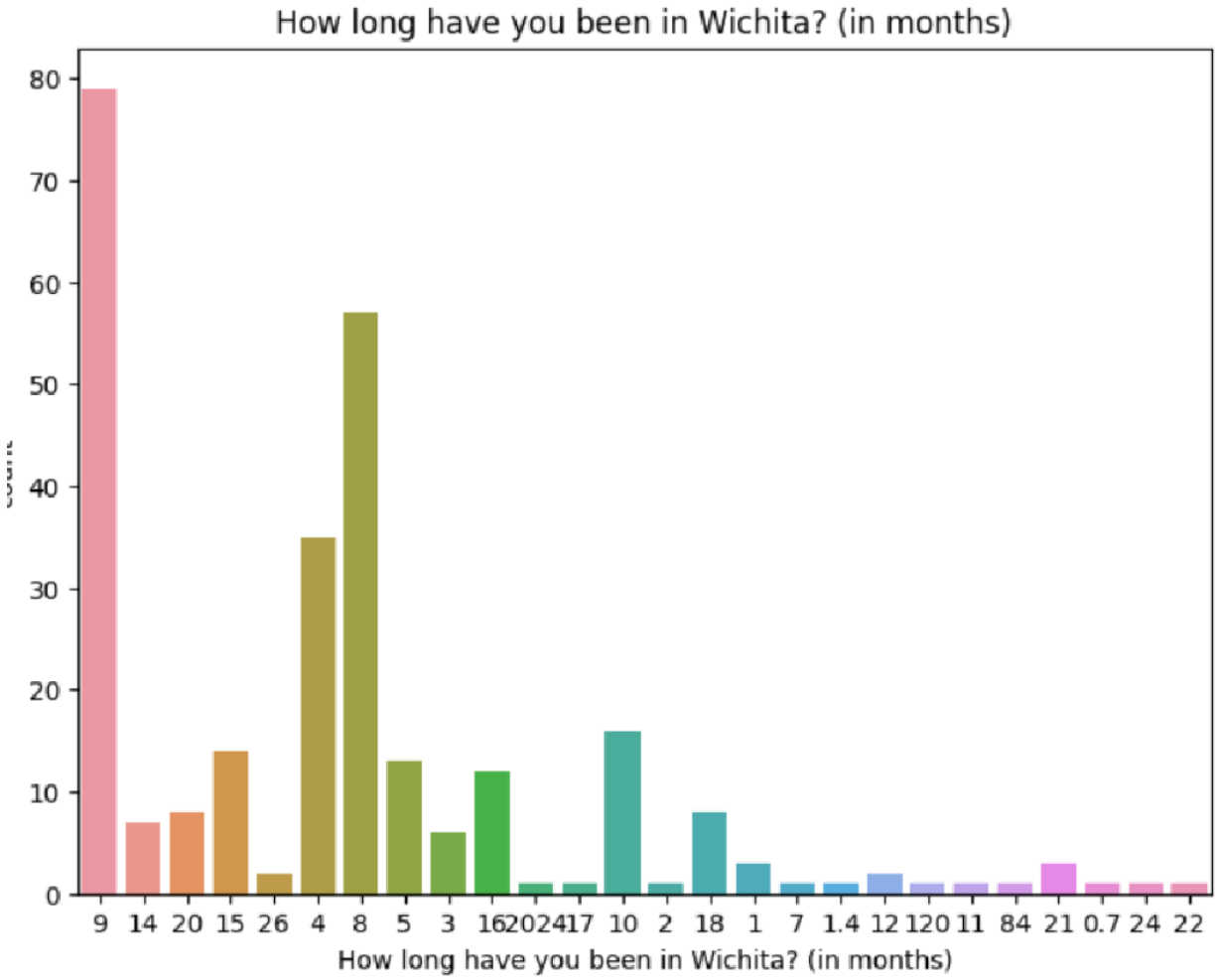
The values generated above will allow our model to predict the data to which dataset it belongs, and based on this accuracy, our result will improve.

The elow graphs which are shown below are generated based on the different datasets. On viewing the chart, we will have a good understanding of which basis the people are getting satisfied with their everyday living.

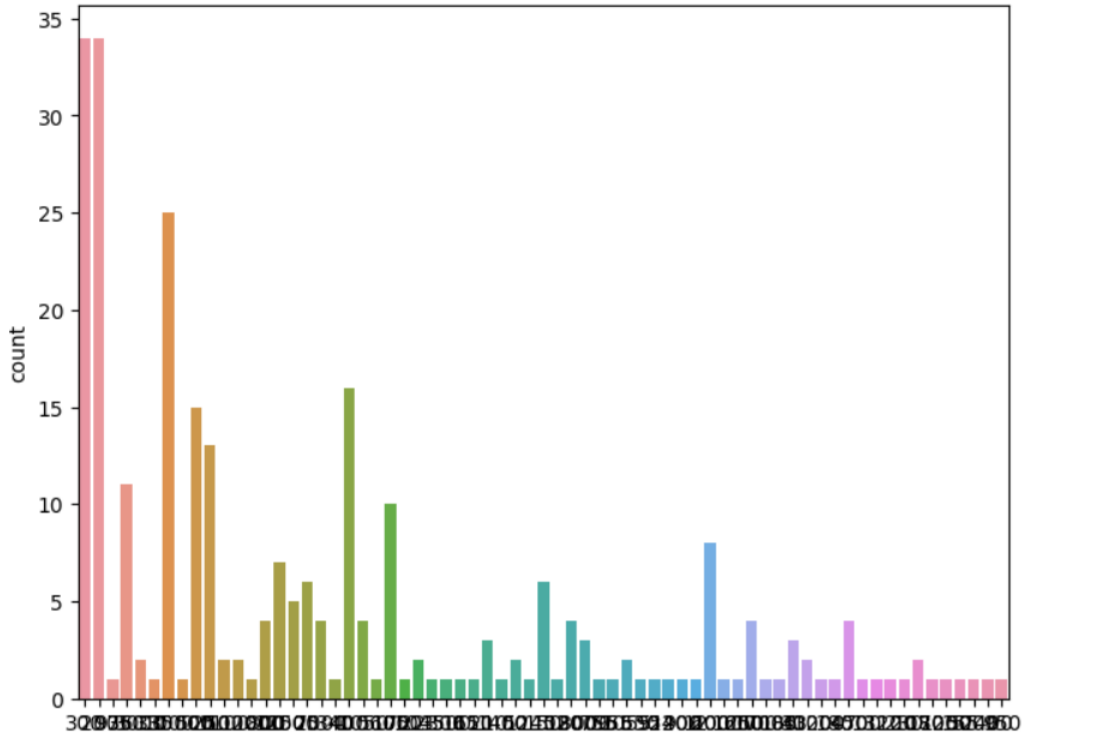

What is the total monthly rent of your unit?
Count
0
10
20
30
40
50
250
500
750
1000
1250
1500
1750
What is the total monthly rent of your unit?

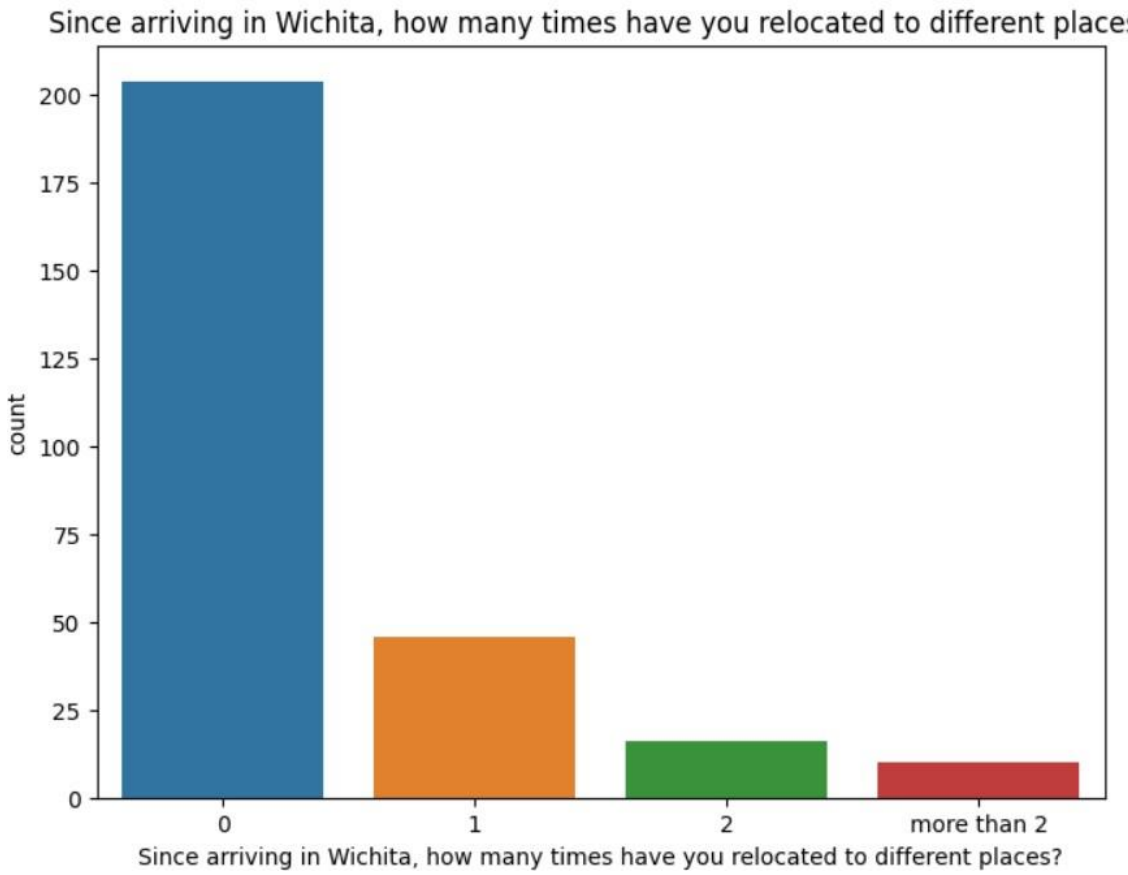

Number of Amenities Paid for by Rent
Count
Number of Amenities Paid for by Rent
0
10
20
30
40
50
60
70
1
2
3
4
5
6
7
8
9


Correlation Matrix
Overall, how satisfied are you with your current living?
On a scale of responsiveness, how efficient is the management of your apartment complex? (in maintenance services etc.,)
What are your thoughts on the level of safety in the vicinity of your residence?
What is the total monthly rent of your unit?
What is the approximate duration of time, in minutes, it takes for you to reach your closest grocery store?
Number of Amenities Paid for by Rent
1 0.69 0.58 0.13 -0.17 0.08
0.69 1 0.68 0.12 -0.091 0.086
0.58 0.68 1 0.13 -0.11 0.052
0.13 0.12 0.13 1 -0.037 -0.12
-0.17 -0.091 -0.11 -0.037 1 0.088
0.08 0.086 0.052 -0.12 0.088 1
Overall, how satisfied are you with your current living?
What is the total monthly rent of your unit?
Number of Amenities Paid for by Rent
1.0
0.8
0.6
0.4
0.2
0.0



What was the total amount you paid initially, covering fees such as application fees, deposit fees, and others?





So based on this graph, our model will be evaluated. This graph is used as a data analysis to assess the result.

Evaluating the Model

We evaluated the performance of the model using the test set. The model achieved an accuracy of 78%, with a precision of 77%, a recall of 78%, and an F1 score of 77%.

Random Forest Classifier

Our analysis shows that a random forest classifier can be an effective model for predicting a person's satisfaction level with their living situation.

The benefits of using a random forest classifier include the following:

- It is a reliable method for dealing with missing values, unpredictable data, and insignificant characteristics.
- Both classification and regression tasks are manageable.
- It can offer a ranking of feature importance that can be used to determine which features have the most influence.
- It can handle massive datasets and is computationally efficient.

Conclusion

The Random Forest Classifier had the best performance among the three classifiers, with an accuracy of 89%. It is a powerful machine-learning technique that can be applied to regression and classification tasks. It can handle large datasets and produce accurate predictions by integrating many decision trees. It is a well-liked option for real-world applications because of its capacity to handle noisy and missing data.

Utilizing measures for precision, recall, and F1-score, the classification report was used to assess each classifier's performance. These are the outcomes:

Results for Decision Tree classifier:

	precision	recall	f1-score	support
1.0	0.95	1.00	0.98	20
2.0	1.00	1.00	1.00	18
3.0	0.81	0.77	0.79	22
4.0	0.80	0.76	0.78	21
5.0	0.73	0.76	0.74	21
accuracy			0.85	102
macro avg	0.86	0.86	0.86	102
weighted avg	0.85	0.85	0.85	102

Results for Random Forest classifier:

	precision	recall	f1-score	support
1.0	1.00	1.00	1.00	20
2.0	1.00	1.00	1.00	18
3.0	0.89	0.73	0.80	22
4.0	0.79	0.90	0.84	21
5.0	0.82	0.86	0.84	21
accuracy			0.89	102
macro avg	0.90	0.90	0.90	102
weighted avg	0.90	0.89	0.89	102

Results for Random Forest classifier (Gradient Boosted):

	precision	recall	f1-score	support
1.0	1.00	1.00	1.00	20
2.0	1.00	1.00	1.00	18
3.0	0.82	0.82	0.82	22
4.0	0.81	0.81	0.81	21
5.0	0.81	0.81	0.81	21
accuracy			0.88	102
macro avg	0.89	0.89	0.89	102
weighted avg	0.88	0.88	0.88	102

Future Work

Although this project is mostly complete, to push it to the next level, the team believes that more data collection will be paramount for the model's ability to become more accurate and generalized for a real-world application. Additionally, a deeper understanding of the various features within the dataset and optimizing our feature selection processes to minimize the risk of overfitting will be essential moving forward. The model may still be struggling with this when predicting certain classes (due to a combination of minimal data availability and feature over-dependency).