AI Boot Camp Project 2

House of Hope

Team Members:

Mathew Watkins

Raymond Dunavant

Terry Hood

Project Overview

Project Purpose / Description

Our project group is focused on enhancing the success rates of addiction / recovery treatment programs utilized by facilities such as the House of Hope. We will begin by identifying and analyzing key factors that contribute to higher or lower completion rates. With current data showing that only 42.1% of patients complete their treatment program, our immediate goal is to develop a model to help improve predictions of client success rates based on an intake questionnaire. Future goals would include training a model to recommend strategies that can significantly improve patient retention and success, thereby increasing the overall effectiveness and impact of these crucial treatment programs.

Project Overview

Goals to be addressed

- Create Model to train Addiction_Recovery dataset(s)
- Create Intake Questionnaire
- Run questionnaire micro-dataset against trained model for predicting recovery success
- Increase Patient success rate above 20%

Future Goals

- Create Outtake Questionnaire
- Create / Train model to recommend strategies that can significantly improve patient retention and success

Project Overview

Approach taken to achieve goals

1. Clone this repository to your local machine or download the source code.
2. Set up the required environment by installing Python and necessary libraries such as pandas, scikit-learn, and tkinter.
3. Develop machine learning models using suitable algorithms for classification tasks.
4. Implement the intake questionnaire interface.
5. Run the questionnaire micro-dataset against the trained models to predict program success.
6. Fine-tune model hyperparameters to enhance performance.

Project Overview – SAMHDA Dataset

Overview of data collection, cleanup and exploration process

- ***Data Loading and Preparation:*** Data is loaded, cleaned, explored and split into training and testing sets.
- ***Model Evaluation:*** The best model is evaluated on the test set to determine its accuracy.

1. LinearRegression
2. KNeighborsRegressor
3. RandomForestRegressor
4. ExtraTreesRegressor
5. AdaBoostRegressor
6. SVR

- ***Cross-Validation:*** Cross-validation is used to ensure the model's performance is consistent and reliable.
- ***Hyperparameter Tuning:*** RandomizedSearchCV is used to find the best hyperparameters for a RandomForestClassifier.

Result/Addiction_Recovery Dataset

Overview of Data Collection / Cleaning

I started with the tedsd_puf_2019.csv dataset that I got from the SAMHSA web site. Privacy policy linked below:
https://www.samhsa.gov/about-us/website-policies-notices/privacy

What made the TEDSD dataset unique was the inclusion of the "REASON" column which we could use as a measure of success given that it gave the reason why the case individual either completed the program or did not.

The TEDSD dataset included the results of over 1.7 million people that received treatment during 2019. This required the use of GIT LFS right out of the gate.

There were not any empty cells to fill but there were a lot with the -9 value, indicating that the information was missing for that case ID.

Result/Addiction_Recovery Dataset

Overview of Data Collection / Cleaning

Working with the directors of the House of Hope Addiction Recovery program out of Wisconsin, we were able to review the dataset and clean it according to what they saw as averages for people that had attended their program.

The primary difficulty they were having is that their program saw only about 20% success with a lot of returning clients

Using the cleaned dataset, we were then able to run correlation matrixes and some trial and error, we found 13 columns that, when applied to the cleaned dataset, was able to predict success rates at over 75% accuracy, 55% up from what they reported.

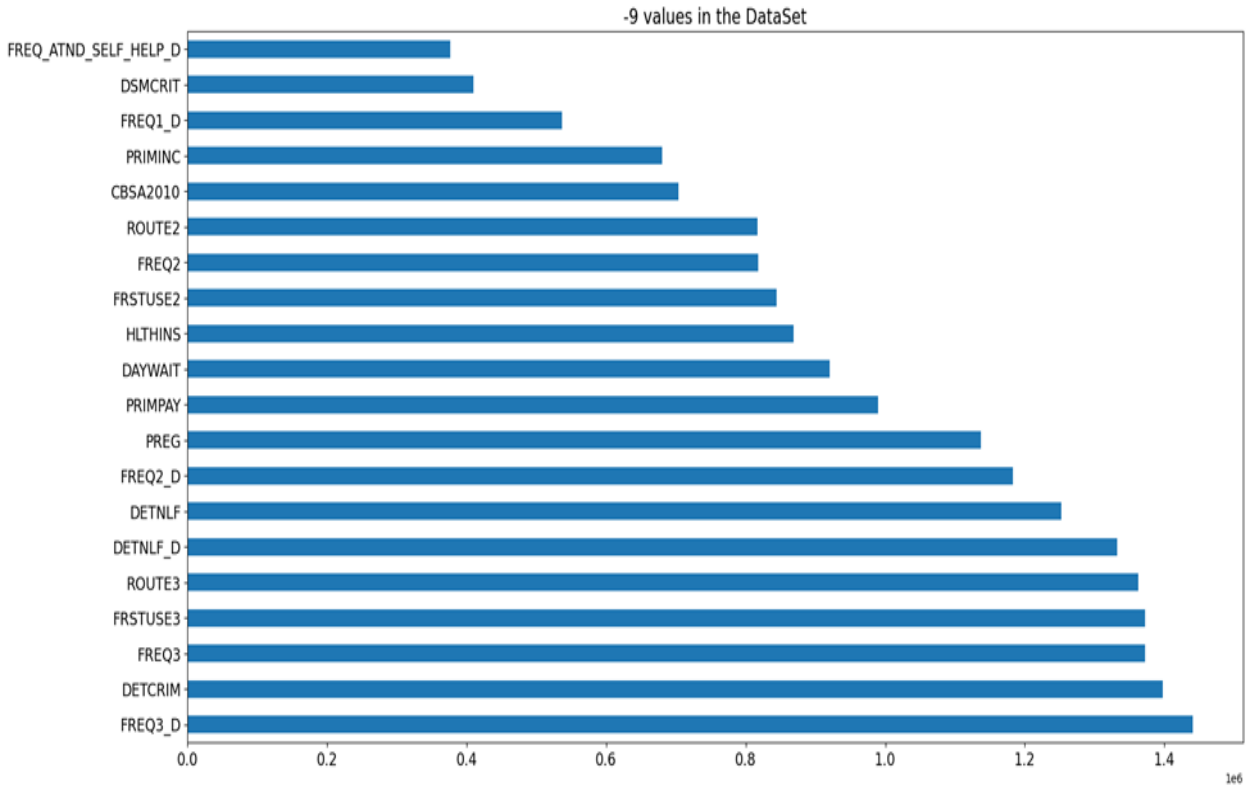
Result/Addiction_Recovery Dataset

Overview of Data Collection / Cleaning - Beau

Locating Missing Values

In our initial evaluation we discovered our missing data was represented by -9 values in the dataset.

We used methods like the **isin() function**, visualizing the missing data, and then like **Feature-Specific checks** to understand the extent of the missing data.



Result/Addiction_Recovery Dataset

Overview of Data Collection / Cleaning - Beau

HANDLING MISSING DATA

The missing data was represented by a ‘-9’
Out of 76 columns there were 47 columns with ‘-9’ missing values.

- We created a new DataFrame and went through the DF and documented all those columns.

In our second meeting with House of Hope we were able to go through each column and place the ‘-9’s in the appropriate Label… Which was also quite the process.

```
# How many columns have -9 values?
print('The number of columns with -9 values:')
print(len([col for col in df.columns if -9 in df[col].values]))
print('We only need to clean 47 columns! Lets get it!')
```

```
# Create dataframe with the columns that hold -9 values in them. Display in a table
missing_values = pd.DataFrame(df.columns[df.isin([-9]).any()], columns=['Columns'])
print('The columns with -9 values:')
print(missing_values)
```

```
df1['FREQ1_D'] = df1['FREQ1_D'].replace(-9, 1)
df1['GENDER'] = df1['GENDER'].replace(-9, 0)
df1['RACE'] = df1['RACE'].replace(-9, 7)
df1['ETHNIC'] = df1['ETHNIC'].replace(-9, 4)
```

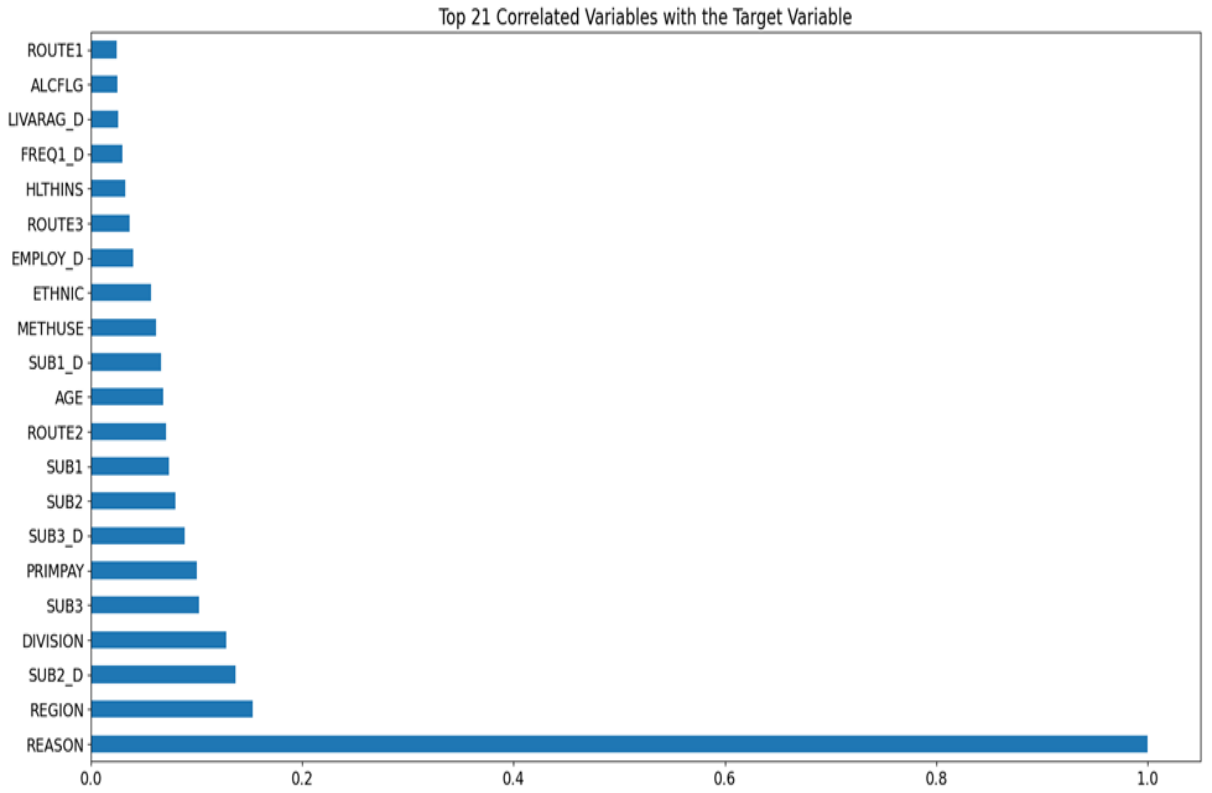
Result/Addiction_Recovery Dataset

Overview of Data Collection / Analysis - Beau

CORRELATION MATRIX

With the cleaned data, we were able to begin looking for correlations in the data relating to the ‘REASON’ column.

After looking for these correlations we were able to find the top 20 highly correlated columns.

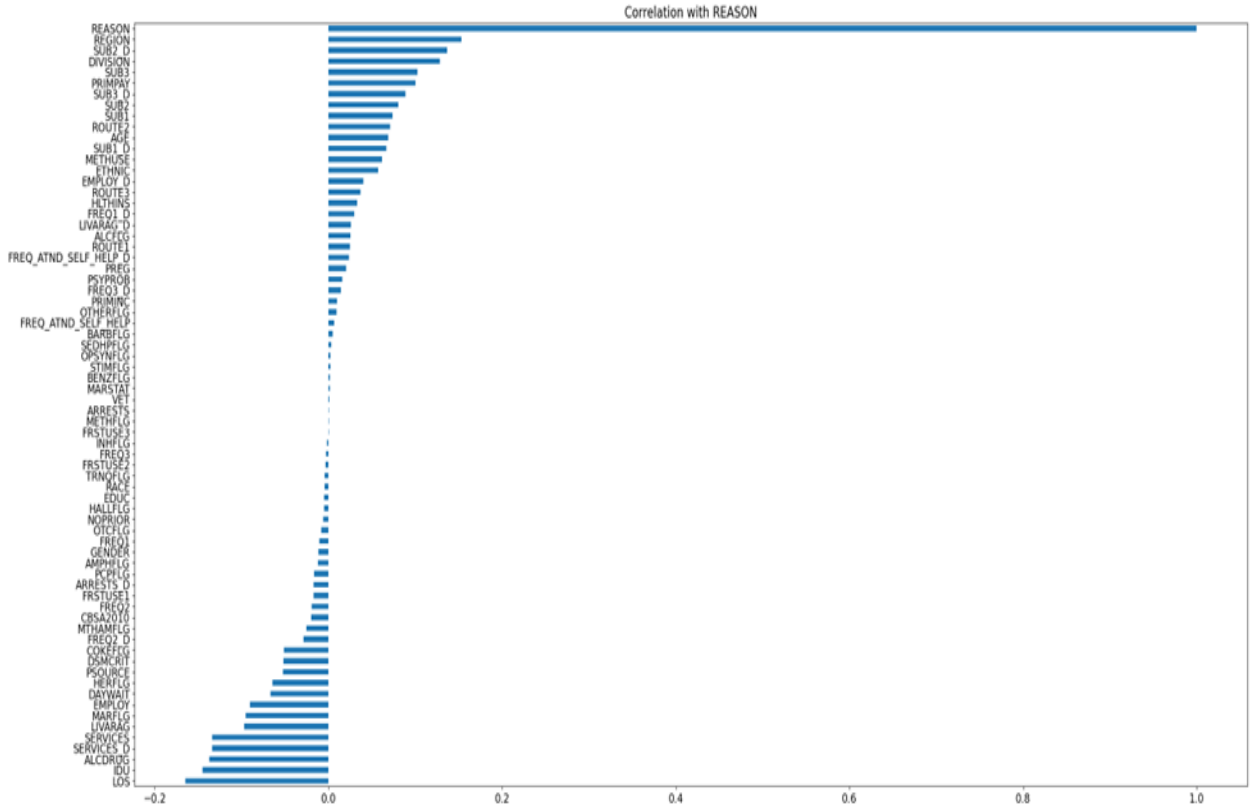


Result/Addiction_Recovery Dataset

Provide a summary of the results with the supporting visualizations (at least 2 per question)

CORRELATION MATRIX

From there we built our correlation matrix to then use that data to begin building and training our model, both independently and together.



Result/Addiction_Recovery Dataset

Model Analysis and Exploration

From the cleaned dataset and a lot of trial and error the qualities that have the greatest impact were Marital status, employment status, living arrangements, services they were receiving, when they first used, how committed they were to attending self help programs afterward, the primary source of payment for the program, what census division they were in, pregnancy, and weather or not an opiate assisted program was used.

Result/Addiction_Recovery Dataset

Model Analysis and Exploration

A lot of these make sense, whereas some columns that seem to only lower the accuracy of the model surprised us like number of arrests, primary income, length of stay in the program, or even what drug or drugs they were addicted too.

Result/Addiction_Recovery Dataset

Model Analysis and Exploration

It was nice to see that race, age, gender and most other factors that you really can not control about yourself, do not see to be limiting factors if you are seeking recovery. All these factors only lowered the accuracy of the model.

Results/Addiction_Recovery Dataset

We ran into last minute trouble getting the .py file that made the predictions to read the cleaned dataset but the code for it is in the GIT hub.

The code that makes the predictions works and is located at the end of the addiction_recovery.ipynb file.

```
y3 = df2["REASON"].values.reshape(-1, 1)
X3 = df3.copy()
# X3.drop("REASON", axis=1, inplace=True)
X_train = df2[["MARSTAT", "EMPLOY", "LIVARAG", "DAYWAIT", "SERV
X_test = df3[["MARSTAT", "EMPLOY", "LIVARAG", "DAYWAIT", "SERVI
y_train = df2["REASON"].values.reshape(-1, 1)
# y_test = train_test_split(X2, y2, random_state=78)
scaler = StandardScaler()
X_scaler = scaler.fit(X_train)
X_train_scaled = X_scaler.transform(X_train)
X_test_scaled = X_scaler.transform(X_test)
# Create the decision tree classifier instance
model = RandomForestClassifier()
# Fit the model
model = model.fit(X_train_scaled, y_train)

# Making the prediction using the testing data
predictions = model.predict(X_test_scaled)
if predictions == 1:
    print("This person has a high probability of success")
else:
    print("This person has a low probability of success")
```

✓ 2m 40.4s

```
C:\Users\matth\AppData\Local\Temp\ipykernel_16912\2656871235.py:15:
  model = model.fit(X_train_scaled, y_train)
This person has a high probability of success
```

Project Overview – Intake Questionnaire

Overview of data collection, cleanup and exploration process

Patient Intake Questionnaire

This application is designed to collect and save patient questionnaire data for test data to be ran against the trained model.

- Collects various patient information including personal details, treatment information, and substance use information.
- Provides dropdown menus for selecting options where applicable.
- Saves data to CSV files, organized by case ID.

Result/Intake Questionnaire

This application is designed to collect and save patient questionnaire data

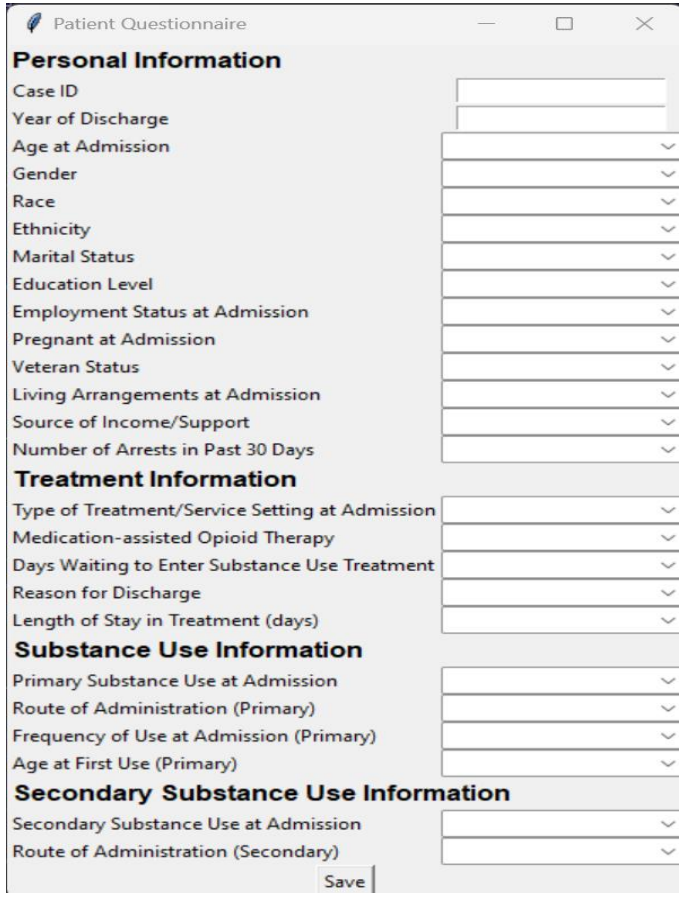
Green County House of Hope Resident Application

House of Hope provides a nurturing, affirming, peer-support environment for women in recovery from substance use disorders. This also entails being able to pursue academic, personal, and professional goals for the purpose of enhancing their quality of life and becoming productive members of society. House of Hope is open to females 18 and older who have maintained a minimum of 30-days of abstinence from alcohol and drugs (unless prescribed) and are actively pursuing a responsible lifestyle and ongoing recovery.

House of Hope is working towards providing a wing for treatment of males 18 and older who have maintained a minimum of 30-days of abstinence from alcohol and drugs (unless prescribed) and are actively pursuing a responsible lifestyle and ongoing recovery.

Result/Intake Questionnaire

This application is designed to collect and save patient questionnaire data



Project Overview – Micro Dataset

The micro-dataset is an output of our patient_questionnaire.py application. The application saves a file to our Patient_Questionnaire_Data folder. The file is read into mathew_watkins.ipynb Jupiter notebook for processing.

Result/Questionnaire Micro-Dataset

Provide a summary of the results with the supporting visualizations (at least 2 per question)

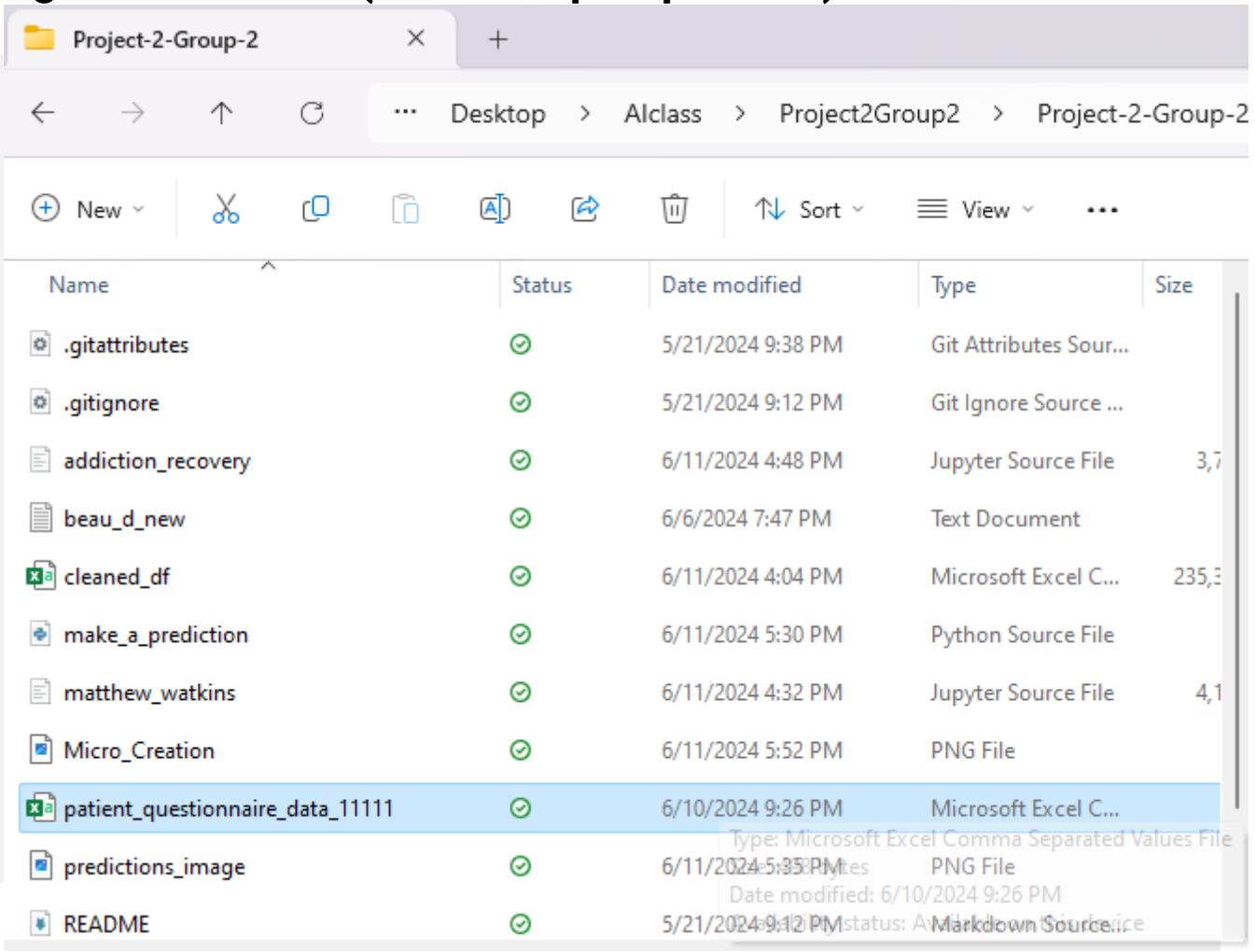
The Micro dataset created by the questionnaire was able to be read in and used as the test column to make predictions

```
def save_data(self):
    # Collect the data
    row_data = []
    for i in range(25):
        entry = getattr(self, f"entry_{i}")
        row_data.append(entry.get())
    # Append to data list
    self.data.append(row_data)
    # Convert to DataFrame and save to CSV
    columns = [
        "CASEID", "DISYR", "AGE", "GENDER", "RACE", "ETHNIC",
        "MARSTAT", "EDUC", "EMPLOY", "PREG",
        "VET", "LIVARAG", "PRIMPAY",
        "DIVISION", "SERVICES",
        "METHUSE", "DAYWAIT",
        "FREQ_ATND_SELF_HELP_D", " LOS", "SUB1",
        "ROUTE1", "FREQ1", "FRSTUSE1",
        "SUB2", "ROUTE2"
    ]
```

Result/Questionnaire Micro-Dataset

Provide a summary of the results with the supporting visualizations (at least 2 per question)

The Micro dataset created by the questionnaire was able to be read in and used as the test column to make predictions



Result/Questionnaire Micro-Dataset

Provide a summary of the results with the supporting visualizations (at least 2 per question)

And here it is being read and used for predictions

```
# now time to read in the results of the questionaire and get our prediction
case_id = input('Enter the case ID: ')
patient_questionnaire = pd.read_csv(f"patient_questionnaire_data_{case_id}.csv")
```

```
y3 = df2["REASON"].values.reshape(-1, 1)
X3 = df3.copy()
# X3.drop("REASON", axis=1, inplace=True)
X_train = df2[["MARSTAT", "EMPLOY", "LIVARAG", "DAYWAIT", "SE
X_test = df3[["MARSTAT", "EMPLOY", "LIVARAG", "DAYWAIT", "SEF
y_train = df2["REASON"].values.reshape(-1, 1)
# y_test = train_test_split(X2, y2, random_state=78)
scaler = StandardScaler()
X_scaler = scaler.fit(X_train)
X_train_scaled = X_scaler.transform(X_train)
X_test_scaled = X_scaler.transform(X_test)
# Create the decision tree classifier instance
model = RandomFore
# Fit the model
model = model.fit(X_train_scaled, y_train)

# Making the prediction using the testing data
predictions = model.predict(X_test_scaled)
if predictions == 1:
    print("This person has a high probability of success")
else:
    print("This person has a low probability of success")
```

(variable) X_train_scaled: ndarray | spmat

✓ 2m 40.4s

```
C:\Users\matth\AppData\Local\Temp\ipykernel_16912\2656871235.py:
  model = model.fit(X_train_scaled, y_train)
This person has a high probability of success
```

Summary

Through the use of AI prediction models, we were able to predict the success rate of patients at a far higher rate than the House of Hope was seeing

Along the way we also gained valuable insight into what was and was not important as far as factors in weather a person can have success or not

We also gained insight into what any person should focus on if they are seeking freedom from any addiction regardless of drug

In the future and with more time I'd like to see if we can manipulate the prediction model to give each individual targeted goals for finding the greatest success

Problems Encountered

Project Challenges

- SAMHSA Dataset was very large (1.7million cases) and required GIT LFS to handle this in our repository
- The dataset has 1,106,580 potential outliers. This significant number of outliers indicates a substantial amount of data points that deviate from the expected range.
- With 76 columns to start with, it took a lot of trial and error to find out what the highest accuracy combination was.
- On the last day, the py file that was supposed to make the predictions would not read a csv in the same folder as it

Future Considerations

Project Discovery - House of Hope

Questions:

- Out of 76 columns in the Dataset, which were meaningful to out project and which ones were noise

Future Development

- *Flask GUI Development* - create Intake and Outtake questionnaire
- *Intake Questionnaire Refinement* - tune the questionnaire to House of Hope changing needs.
- *Outtake Questionnaire Creation* - to gather data regarding client experience to gain insights into what works well within the rehab sessions.
- *New Model Development* - training a model to recommend strategies that can significantly improve patient retention and success, thereby increasing the overall effectiveness and impact of these crucial treatment programs.