**Cocaine Usage Prediction Program**

**Goal of the Project**

The Cocaine Usage Prediction Program was made with the goal of monitoring the usage of cocaine throughout the United States and predicting the future rise of the usage of the substance. The project should be able to use pre-existing data about the usage of cocaine in the United States and visualize the current and future usage of cocaine throughout the country.

**Significance of the Project**

The Cocaine Usage Prediction Program is meant to aid addiction specialists, especially those who specialize in cocaine in specific. The usage of the Drug Abuse Project will allow them to determine in the future what areas need help the most by allowing them to look at statistics such as how many users there may be in a specific state, or by which age demographic is most likely to use cocaine in said state. Allowing them to look into the future of what may happen in certain states may allow them to curb the growth of users in any means necessary.

The Drug Abuse Project does this by use of artificial intelligence, which is done by a model provided by Scikit-Learn. The model in question is the support vector model, which specializes in cases of regression and classification.

**Installation and Instruction on How to Use**

The project will not be able to run unless you have installed specific python modules using pip. Alternative methods are available at the links provided.  
Pandas: <https://pandas.pydata.org/docs/getting_started/index.html>Scikit-Learn: <https://scikit-learn.org/stable/install.html#install-official-release>Matplotlib: <https://matplotlib.org/stable/users/getting_started/>Portalocker: <https://pypi.org/project/portalocker/>

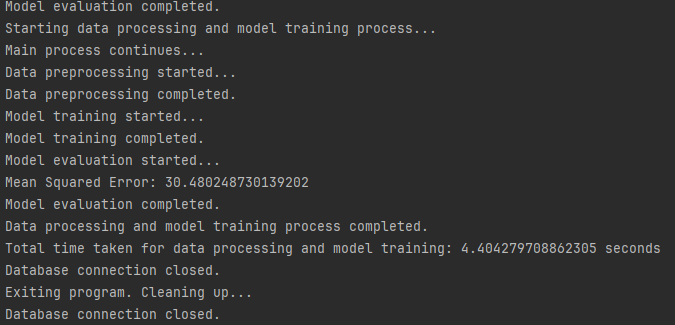
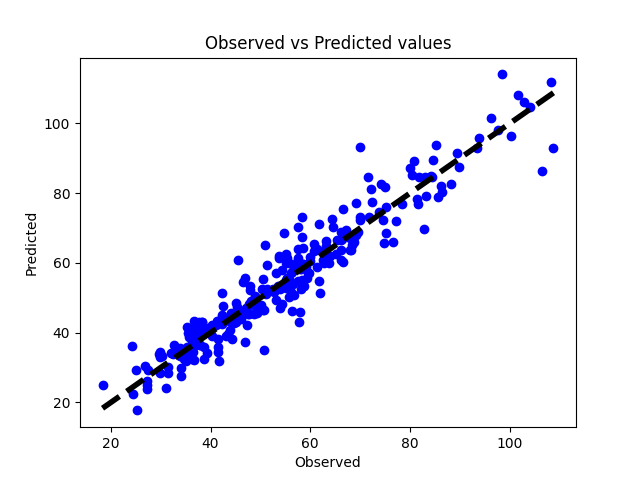
After the modules have been installed, the project can be run. It creates a csv file that is named “predicted\_data.csv”.

Code Structure

Our code structure is built to read the csv file first, then preprocess the data to clean up.

Then, we process that data and split it into training and testing sets. Our model is a linear regression model that we use to feed the data into and predict trend of cocaine usage in the upcoming future. And then we apply operating systems concepts to the project and demonstrate file management, a form of multiprocessing, file locking, concurrency, and signal handling.

Functionalities and Test Results

The project calculates the mean squared error while predicting the trend in the future usage of cocaine. It also writes the predicted data into a csv file.

Discussion and Conclusion (Includes things like limitations and such)

So, there are some limitations that this project runs into. When it comes to multiprocessing, I wanted to see if I could apply that into processing data but it couldn’t work due to the nature of the csv file and the data that is contained in it. As for locking, it is used excessively like how preprocessing, model training, and evaluation are wrapped in locks, which could lead to unnecessary performance.

As for threading and multiprocessing, they are combined to achieve a result when it creates separate tasks for data processing and model training. But it can lead to issues with synchronization and resource management.

With signal handling, it only handles the termination signal. If we had more time, we could expand on the concept of signal handling.

I believe we were able to implement the concepts of operating system structures and can be improved and expanded overall.