Deliverable 2

1. Problem statement

This project will try to model the recidivism of offenders released from jail based on multiple factors such as initial offense class and type, characteristics of the offender and release type.

2. Data Preprocessing

I am still using the dataset https://www.kaggle.com/slonnadube/recidivism-for-offenders-released-from-prison which has about 26000 records and 12 features in the first file and 6718 records and 16 features in the second file. As these data is well structured and categorized, I didn't do much preprocessing. However, I excluded the first two columns which are release year and recidivism year, as they don't play a significant role in recidivism. As I'm concerned with whether an offender will reoffend, the last column, which tells if the person has recidivism, becomes the most important classification feature.

3. Machine learning model

I used LinearSVM and Random Forest model which we learned from the assignments. I didn't use Gradient Boosting since we haven't learn it yet, but I might implement it in the future. However, Random Forest was already a proposed model in my initial proposal because of its application in multiclass classification, and it proved functional, although the result seems to be overfitting. LinearSVM seems to be overfitting too. Both models gave 100% accuracy.

To implement the models, I basically followed the process in the Assignment 2, and add some other functions that fits my data set.

I haven’t really applied any optimization techniques so far, and the ratio of training test split is also relatively random, which is 0.5:0.3:0.2 for train : test : validation. I will try to manipulate the hyperparameter of my classifier to reduce overfitting.

I had some trouble trying to vectorizing my dataset and visualizing the result, but after some research on the internet and apply similar techniques as in the assignment, I solved them.

4. Preliminary results

My models generated 100% accurate results on test and validation set, while the training is of course conducted on the training set. But I have no idea why this is the case, maybe they just perform too well. I implemented sklearn’s SVM and Random Forest models.

5. Next steps

I will try to set different hyperparameters to see if they will generate different results.