I am working on what Machine Learning Algorithms we can use for Forecasting the Crypto Coins we are using in this project . As the first leg of the project we had to decide on what algorithms would work on the forecasting in real time and I decided that the Machine Learning algorithms we would like to use would be three . Upon working on the project we will decide which algorithm will work better in real time and has the least mean square error as that is the criterion we are basing the forecasting on along with the accuracy . So the algorithms are Random Forest in this particular case it would be a regression random forest because we need to forecast the next return . The second one would be SVM. So we know that the SVM ‘s can be used as both classifiers and Regressors but here we are using it as a regression algorithm.Last but not the least we would use LSTM.We will train LSTM with a data that will be suitable to us and we can find sustainable methods for prediction . Let’s go into the three algorithms in detail.

1)RANDOM FOREST

I chose random Forest regression primarily due to the fact that it is an ensemble learning method which means it combines multiple ML algorithms for prediction, and outputs the best performing method. Since the dataset is large enough, I opted for random Forrest’s default split of 75:25. Also, since bitcoin data is continuous, a regression model was chosen.Basically in RF we have a regression tree. To forecast we would need to study the tree from the node and after that we would make a series of tests and certain branches will be chosen from the terminal node and this defines the value that we use for forecasting the needed dependent variable .Which means if the coins price might increase or decrease and following time period .aSo for each node we need to take a random subset of the dataset and train the dataset based on the observations we define the choosing of the branch .RF forecasts are then basically obtained and then we would average the forecasts based on different trees.

2)SVM

Support vector machine that we are using here follows the regression model.It tends to have a lot of feature spaces.This machine learning algorithm has a lot of kernels which can be used to predict or forecast the prices of the cryptocoins .The reason for this prediction is the fact that SVM uses hyperplanes.What these hyperplanes do is they tend to enlarge the output margins hence preventing overfitting .Now we know that the data especially financial data when plotted in the graph is usually linear hence for this reason we are using different kernels of SVM called Support Regression (SVR)model. We use the RBF which is called the Linear , Polynomial and Radical Basis Function kernel for forecasting the major cryptocurrencies.We will look at the popular cryptocurrencies and decide the main ones . So upon my research I found out that the data is in time series and the cryptocurrencies use bivariate time series so for that there would be a continuous dependent variable which is the daily closing price for each crypto and MSCi which is the daily close price as the predicting variable which is what we would consider in this project as well.The criterion we are taking will be the RBF mean square kernel and RMSE which is the root mean square and we would show the best working algorithm too.

3)LSTM

For the financial data forecasting LSTM nn makes an excellent algorithm for the patterns we would look for and the need to extract suitable patterns for modeling and the prediction.For this we need to install a few python libraries and import them.For this the best way to obtain the time series dataset is by using an api or the time series API.For the LSTM model to work we need to perform the data analysis using EDA that needs the open and closing prices .After the EDA we can then properly function using the LSTM model and that for the close price prediction .We then continue to so the data pre-processing and training and testing spits.We need to define the model and its respective layers .For this usually we use the 10 hidden layers with relu function and the dense single layer.The loss function here computer the quality of the model.We use the LSTM regression analysis for this .

II) What I did this week

What I worked on this week was using LInear Regression model on the financial data but upon using it on the time series I realised that this model started over-fiitng and cause a huge RMSE.

I am explaining it below.
One of the models I am using for bitcoin forecasting is Linear Regression. It is basically used for relating or correlating variables and forecasting. The reason I chose Linear regression for bitcoin price forecasting is because using regression analysis is widely used and widely applicable. This analysis will give you a better understanding of the statistical inference overall. To begin with we are importing the dataset from bitcoin_tweets_2021_prices_final (1).csv.The dataset contains 6 features which are data,text,user_followers,user_favourites,compound,open,close and volume. These features were obtained by merging the bitcoin data and the tweets. I then perform data analysis which is exploratory and then describe the data which is seen as count,mean,std and min. I then check for null values in the feature dataset but thankfully due to data-preprocessing there were no null values present in the features. Then I move on to correlation , the reason we do correlation is to see the relation on how one or more features are basically related ,I do so by using the corr() command for seeing the correlation of all the features. After that we are going to visualize the correlated features using both corr() and heatmap() of the seaborn library. Now I will implement correlation() basically take the data and the threshold value with basically 0.81 value and display the returned features .Here our target feature is ‘Close’ .So now the correlation of the target feature with column is ‘Close’ and ‘Open’. Now we need to separate the independent and dependent features .Splitting the data into training and testing and for that we use the train_test_spilt function, checking the shapes of all the four features .Then I apply feature scaling basically scales the data in a range for applying the normalization technique to normalize data. Then I test the model using test data. Then evaluate the model that it is performing .Checking the mean absolute error for finding out the difference in the price range and the error came out to be 1375.2581805400414. Similar process was done for Elon Musk’s Tweets . Here the dataset and the tweets are really less and merging the datasets to the price forecasting did not produce much results because of the quantity of

tweets . The error however is less comparatively but it's still high , one can say looking at this that the price fall from day to day is less in correlation to how often Elon Musk tweets and how people get influenced by it . But it is not significant. Mean Absolute Error = 1511.5784879089504. I also calculated the R scores for both the dataset and for the bitcoin dataset 0.9822261339260688 and for elon musk 0.9625261339268658.