* Urls of the desired time period for each data set are gathered and put into python
* Ats.readTimeSeries is used to download each dataset from the sets of urls
* createFullBase was edited to deal with differing frequencies and is used to combine the different timeseries of various frequencies and make them all the same frequency by separating the shorter frequency data into multiple variables for each occurrence between the largest frequency
* Note this requires that the shorter frequency fits into the larger frequency an integer amount of times
* Note2 this is generally only useful for using the longest frequency variable as the target variable, variables of longer frequencys are discarded.
* CCF, dynamic time warp, and pearsons r coefficient are used to evaluate the best predictors of each variables.
* The data is made stationary by differencing and checked to be stationary before running CCF
* The data in its original form is used for dynamic time warp
* The data is then saved as CSV files and loaded into R.
* For arima the data is first made stationary and checked to be stationary (all data including exogenous variables not just target data)
* The data is then looped through a function called crossvalid which splits the data into k test and training sets based on the HV blocked method. K=5 folds has been used for the testing so far.
* A suitable H blocks of data to be discarded either side of the test set is determined by viewing the ACF and PACF plots to see how many dependent lags their are and a number larger then this is ideally chosen to ensure independence between the training and test sets
* This is not always possible as some datasets have correlation that seems to extend forever, no current plan on how to deal with this, additional differencing does not seem to have an effect.
* An arima model is then generated from the training set and used to produce fitted values from the test set.
* The test set and fitted values are then un-differenced back to their original series to calculate error
* This is looped for different orders of AR and MA models and the model with the lowest error from the 4 error methods is chosen.
* For MLPRegression the data is **not** differenced to be stationary.
* The data is normalised as a whole between 0 and 1
* The same function is used to separate the data into k test and training sets with H lags removed
* The separated data is then saved to CSV and transferred to python
* The appropriate data are then matched in DataFrames (eg the predictor and target variables) to allow null values to be removed by row .
* The data is then separated back out to predictor and target variables and MLPRegresion is run and the fitted values saved.
* The fitted values and corresponding target test values are un-normalised per k fold
* The values are then saved as CSV and transferred back to R
* The same error measures are used excluding relative error.

For Generalised linear models (GLM)

* Data is not made stationary
* A crossvalidation function similar to the arima is used except it generates a linear regression model
* Residual plots for the desired variables are looked at to check if any data transofmation is necacary.