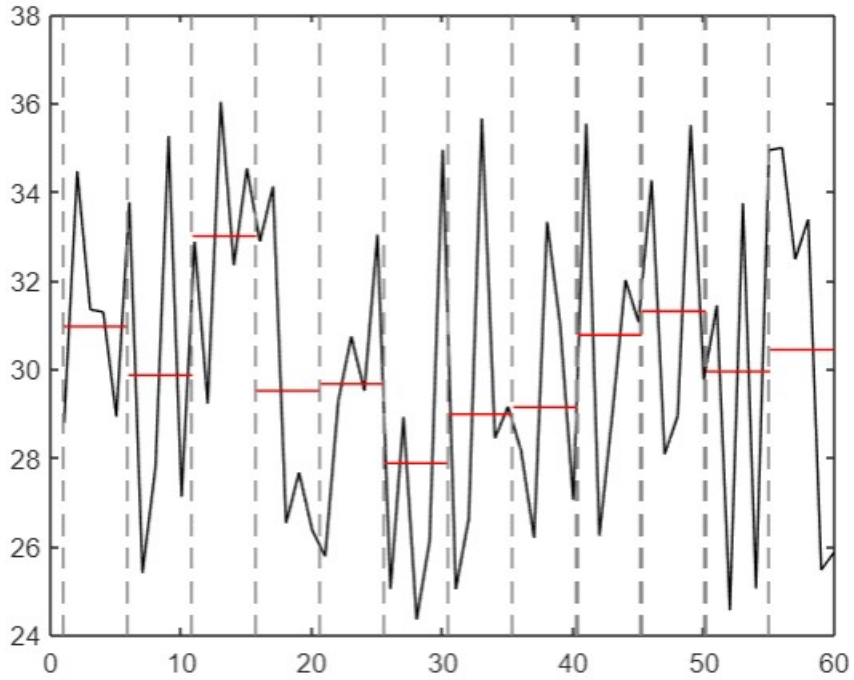
Ryan Bellinger
Dr. Winikus

Time Series Project Final Report
11/1/2021

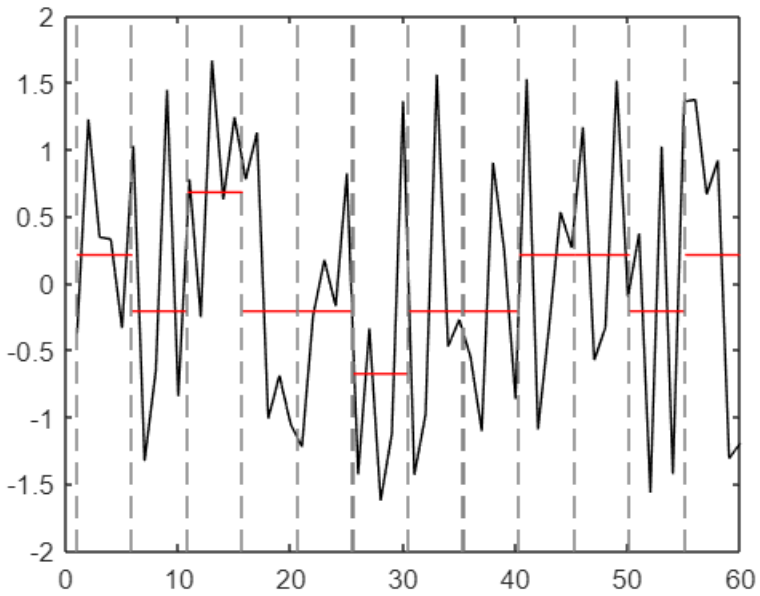
This project combined a fairly large series of elements so they will be discussed in order of implementation in this report.

The first element was the conversion of time series into piecewise aggregate approximations of time series. This was a fairly simple task, as each of the original time series were iterated through and for each predetermined timeframe length the sum of all values within that timeframe was taken. After the timeframe was over the mean of that timeframe was taken and turned into a piecewise function to represent the entire timeframe.



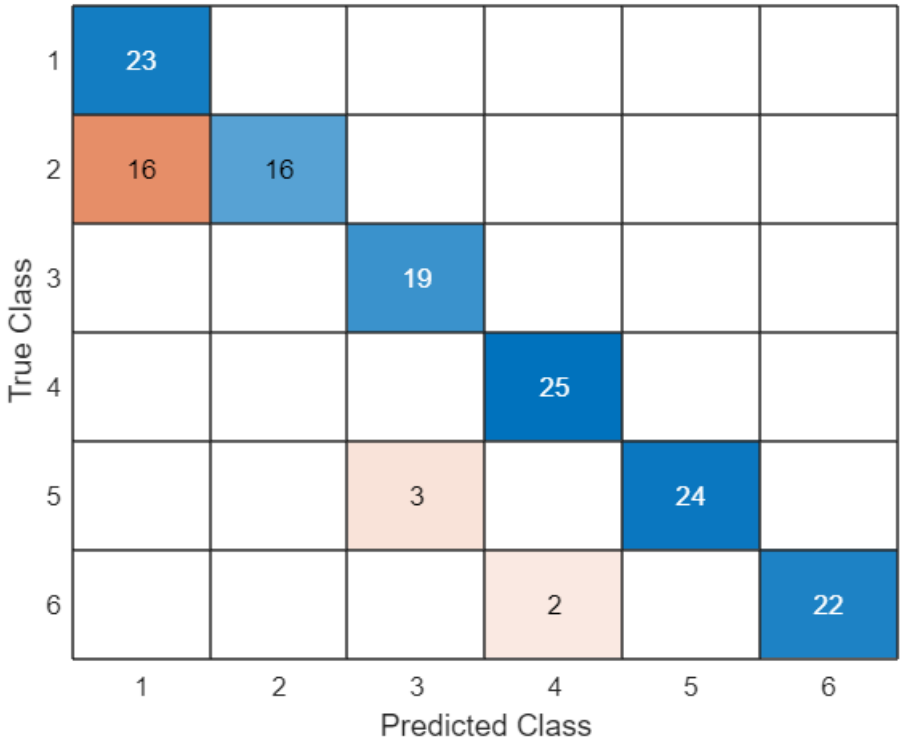
The above represents the output of the PAA process from that timeframe.

SAX was computed much in the same way. First all of the data in the timeseries was normalized, with mean 0 and standard deviation 1. Then, the PAA of this normalized data was taken exactly as described above. Then conditionals were used to assign each resulting value of the PAA a symbol from a to f. 6 letters were chosen for the alphabet because the data needed to be sorted above and below the mean, and then were chosen to represent where approximately 33% of the data would fall after the split was made. To try to neatly get approximately each letter representing one sixth of the data, positive and negative .97 and positive and negative .43 were used as cutting points as well as zero. To represent these letters graphically, a letter would then convert back to around midway through the distribution they were supposed to be representing.

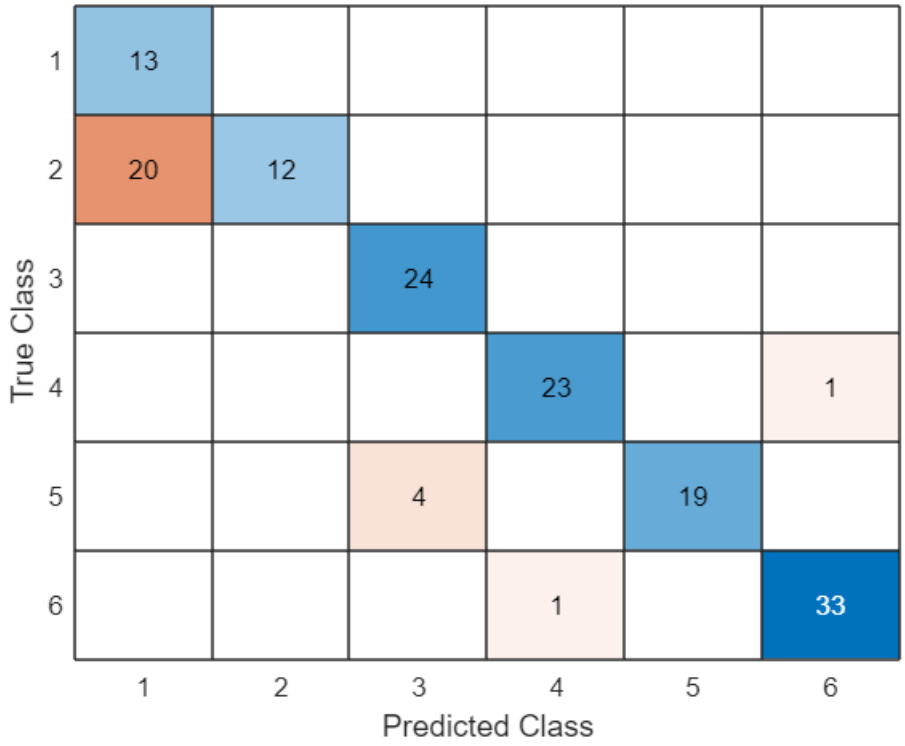


The above represents the sax representation of the same time series.

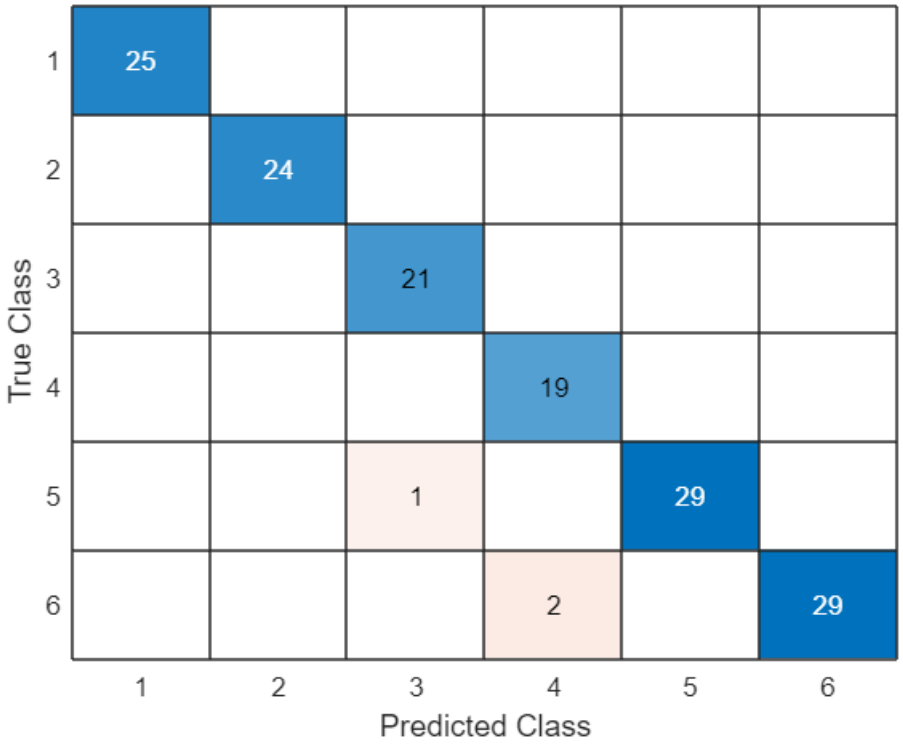
The following are the results from classification using a knn model in the form of confusion matrices. Each is labelled by its perspective parameters.
PAA Series with 8 frames, euclidean distance



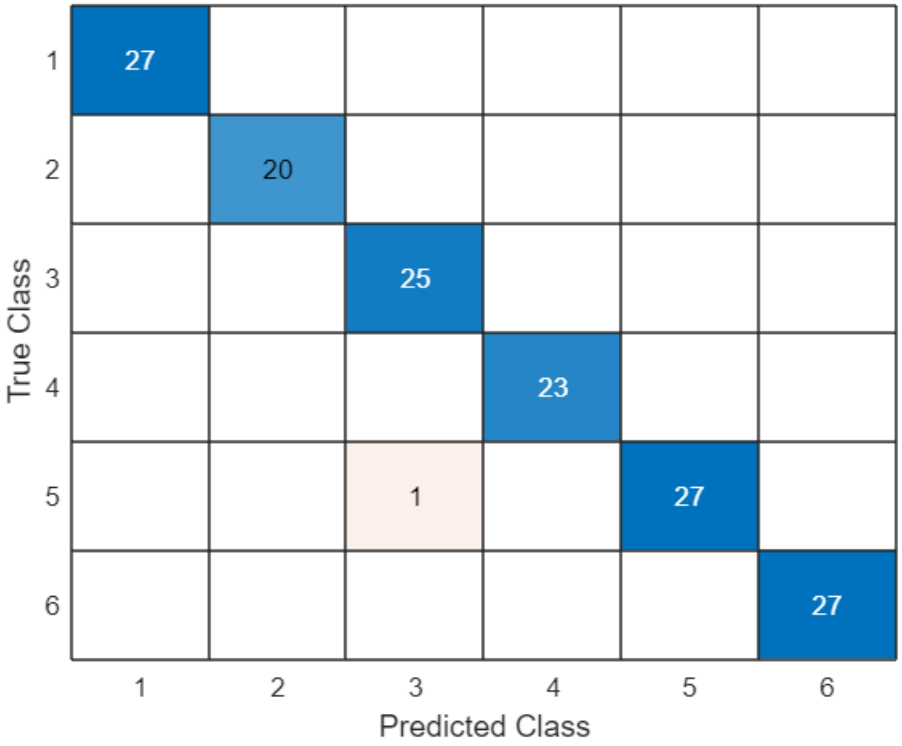
PAA Series, 8 frames, Manhattan distancing



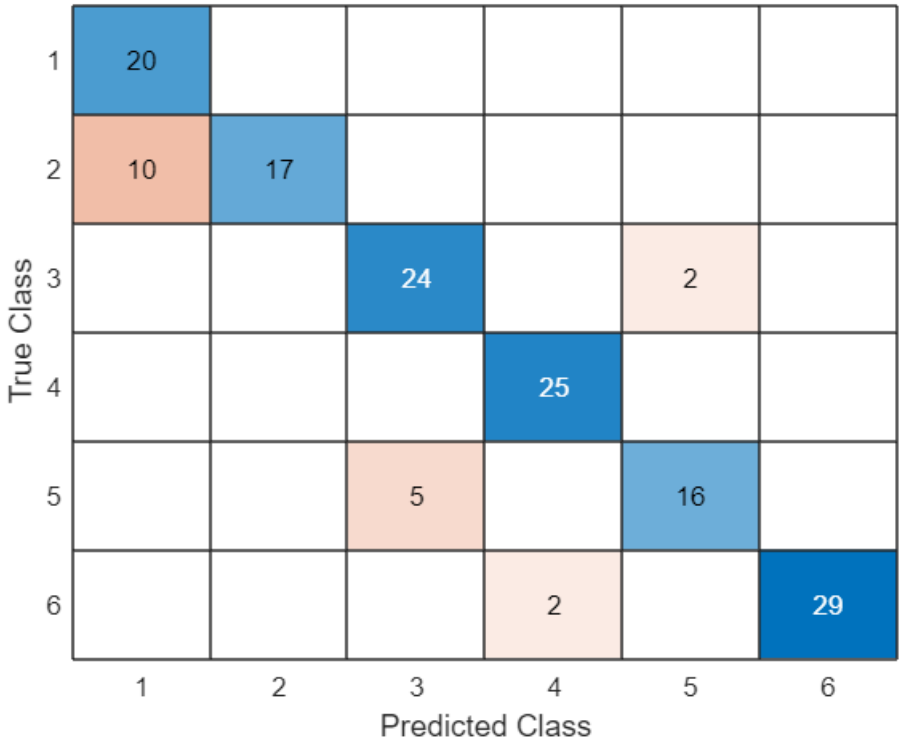
PAA Series, 12 frames, Euclidean distancing



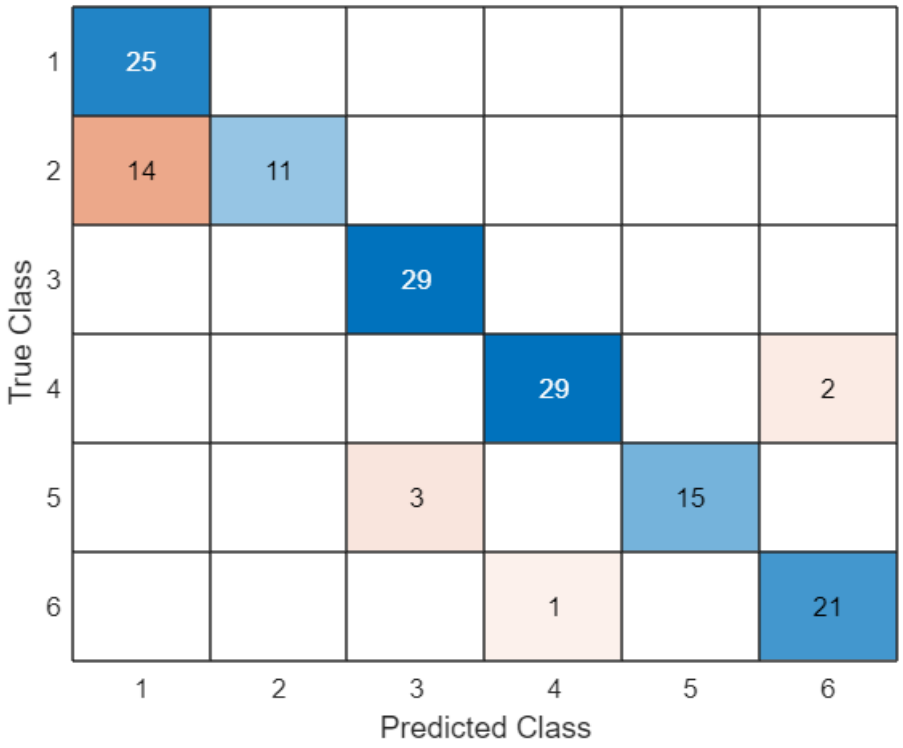
PAA Series, 12 frames, Manhattan distancing



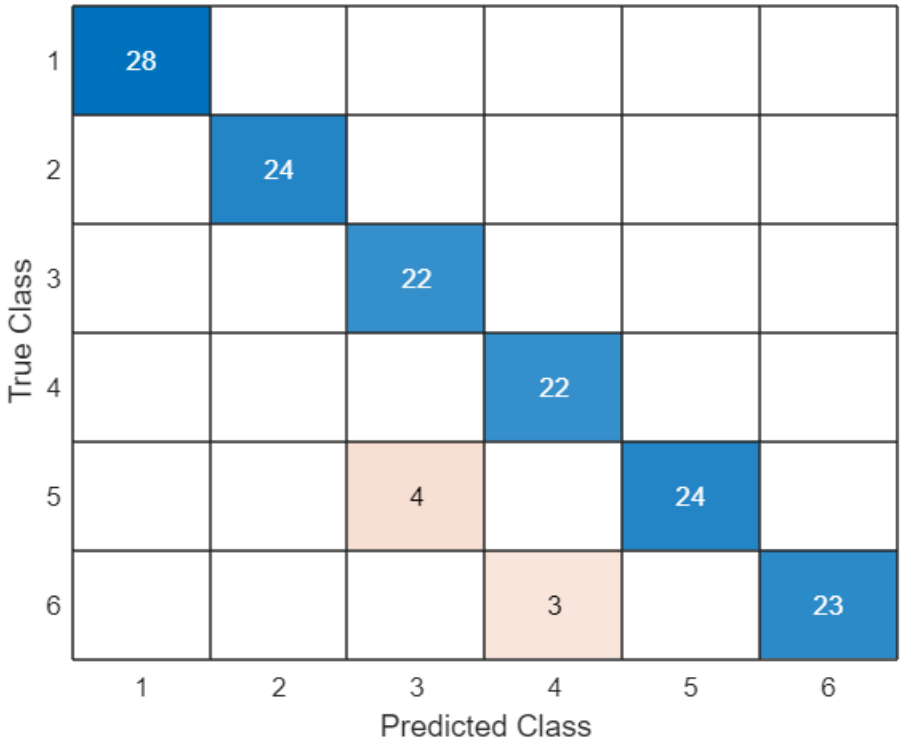
PAA Series, 16 frames, Euclidean distancing



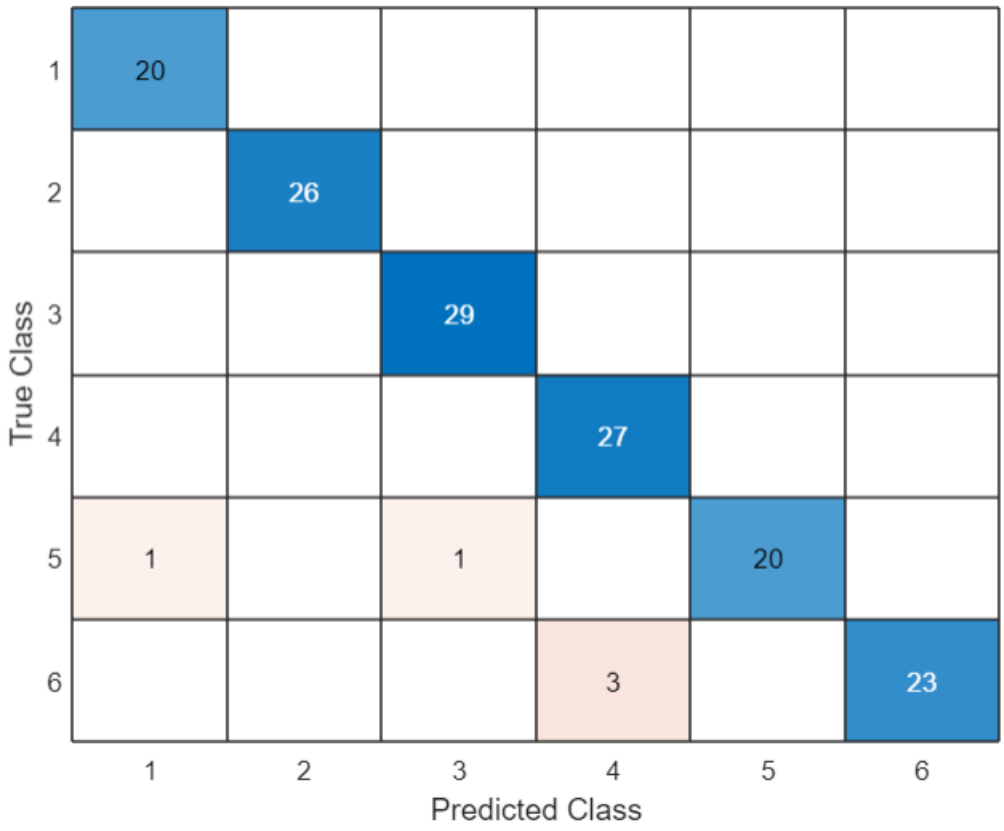
PAA Series, 16 frames, Manhattan distancing



Full dataset, Euclidean distancing



Full dataset, Manhattan distancing



What can be concluded from these results is that around 12 frames is the best amount of data to help the model make the necessary classifications. There is a clear underfitting effect occurring when facing data that has 8 frames, but overfitting also appears to be occurring when there are 16 frames. Interestingly when the entirety of the dataset is used the overfitting effects appear to go away again slightly, as the model well beats its performance on the PAA approximation with 16 frames, but can't quite reach the accuracy that was reached when using the piecewise approximations with 12 timeframes. Neither Euclidean nor Manhattan distancing has an extremely apparent advantage when compared to each other as neither significantly outperformed the other in either situation. Overall, the model reached remarkable degrees of accuracy, the only struggle it really had was differentiating normal vs cyclic data when using too many or too few frames.

The preprocessing to generate the training and testing data was fairly straightforward, using a cvpartition to randomly permute the dataset with 25% of the dataset being held out as training data. It's not the most effective way to create an accurate model, but it aided with the simplicity of generating the confusion matrices.