IL2233 Final Project
Time-Series Prediction and Anomaly Detection

Shawn Nagar

23 May 2024

1 Time-series prediction with neural networks

1.1 Prediction with synthetic series using MLP, RNN, and LSTM

Generate the following uni-variate series, and then use neural networks to do in-sample and out-of-sample predictions

1. An equal-difference series starting from 0, ending to 1 (excluding 1), with a length of 200 points (step = 0.005). Design an MLP for one-step prediction. The output vector has a size of 1. Let the input vector be a size of 4.

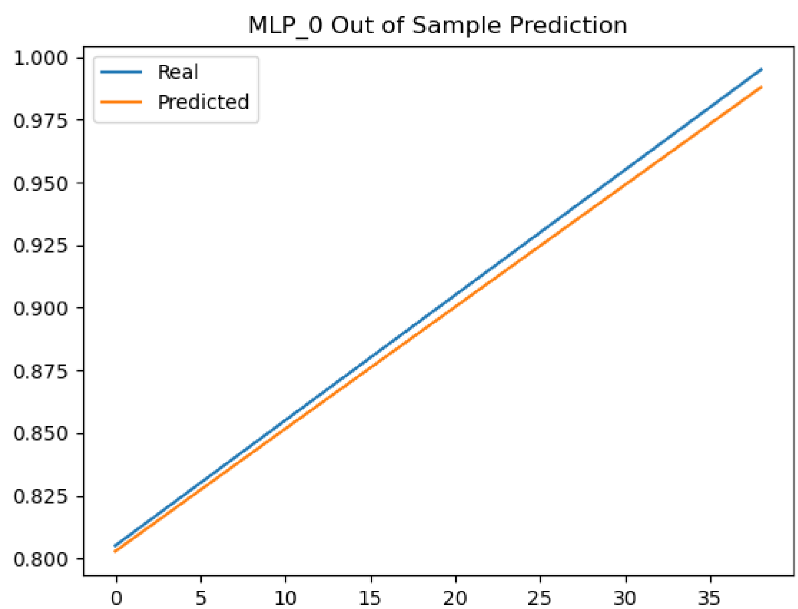

Figure 1: MLP Out of Sample Prediction

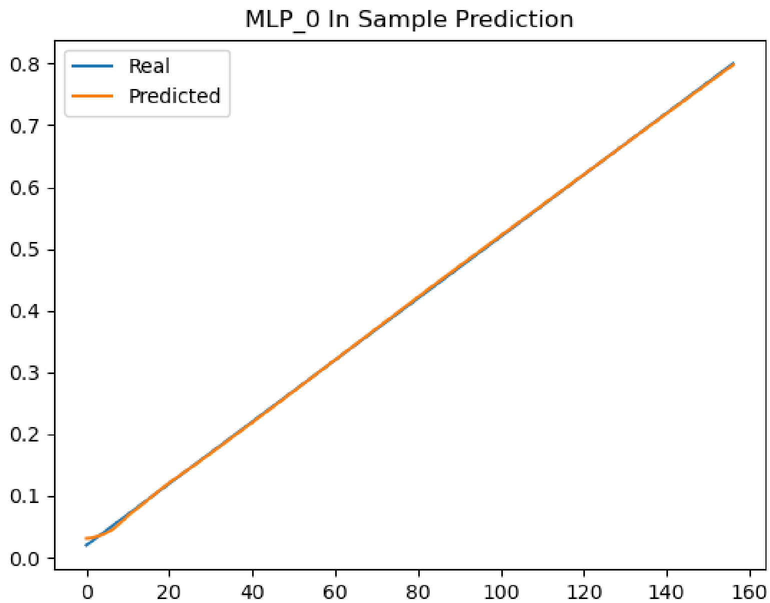

Figure 2: MLP In Sample Prediction

R2	0.988
MAE	0.005
MSE	3.594
MAPE	0.006

Table 1: MLP Accuracy Metrics

2. An equal-difference series starting from 0, ending to 1, with a length of 200 points (step = 0.005), plus white noise i.e., random variable with zero mean and 1 variance. You may need to control the amplitude of the noise series in order to control the signal-noise ratio. Design an MLP for one-step prediction. The output vector has a size of 1. Let the input vector be a size of 4.

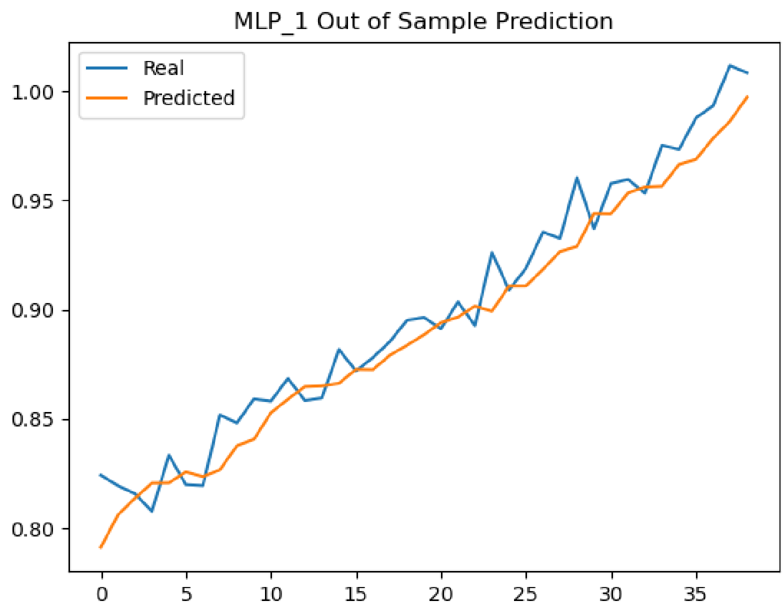

Figure 3: MLP Out of Sample Prediction

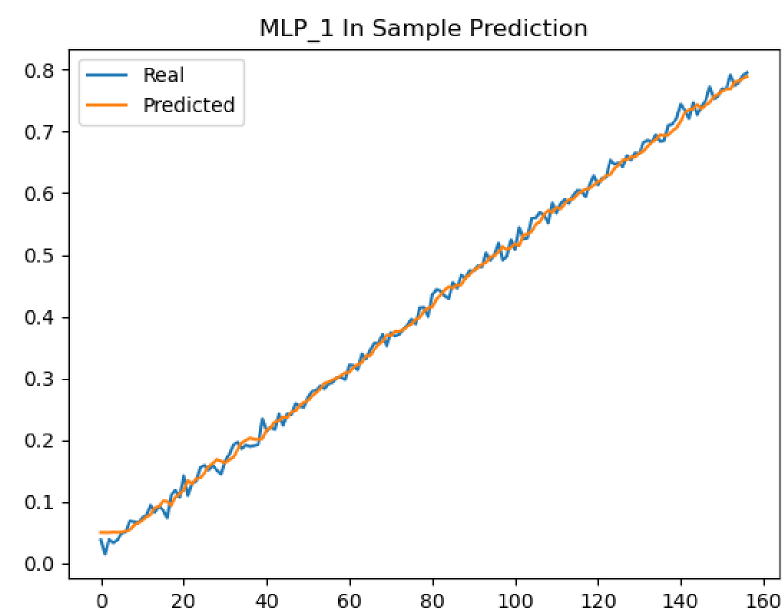

Figure 4: MLP In Sample Prediction

R2	0.934
MAE	0.011
MSE	0.001
MAPE	0.012

Table 2: MLP Accuracy Metrics

3. A deterministic series sampled from a sinusoidal wave with period 20 seconds, with a sample rate of 100 Hz. Generate sufficient samples (at least 3 periods of data) as needed to achieve good performance, e.g. MSE (mean squared error) below 0.5. Design an RNN and a LSTM for two-step prediction. The output vector has a size of 2. Set the input vector size by yourself.

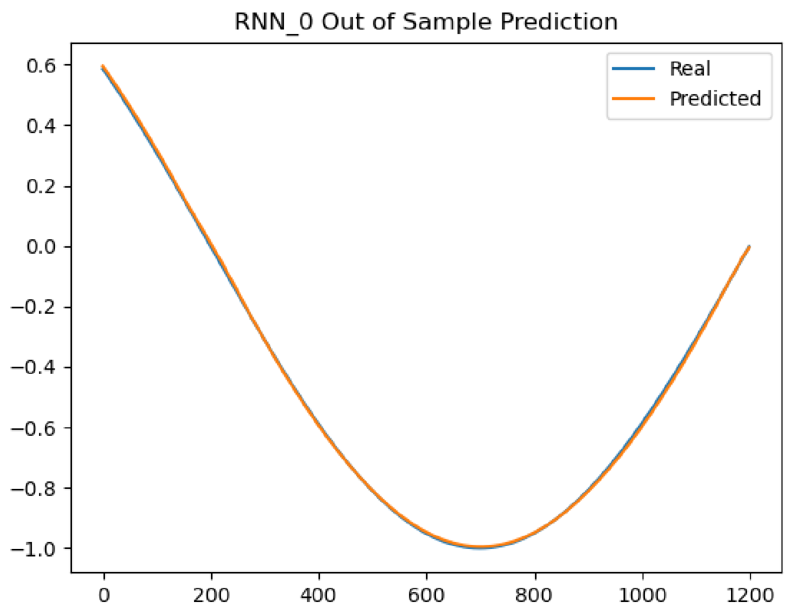

Figure 5: RNN Out of Sample Prediction

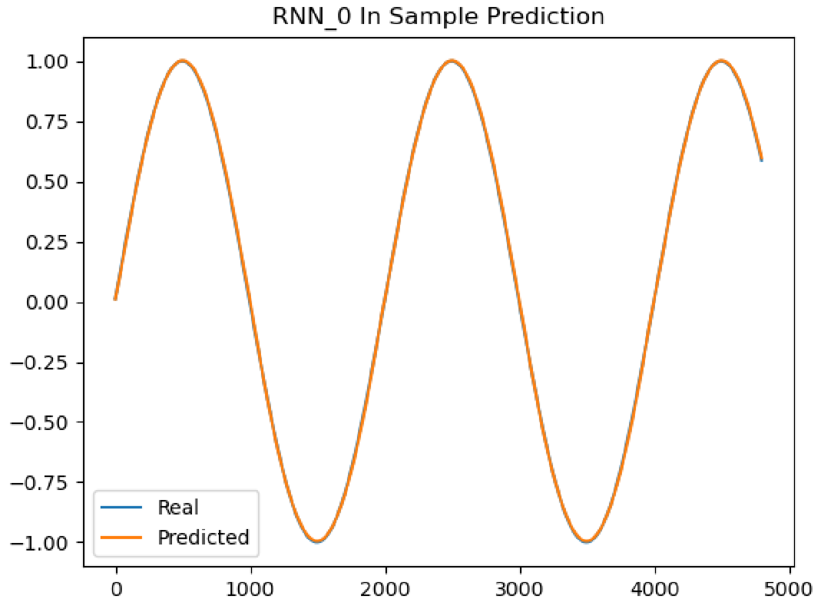

Figure 6: RNN In Sample Prediction

R2	0.999
MAE	0.005
MSE	0.00
MAPE	0.04

Table 3: RNN Accuracy Metrics

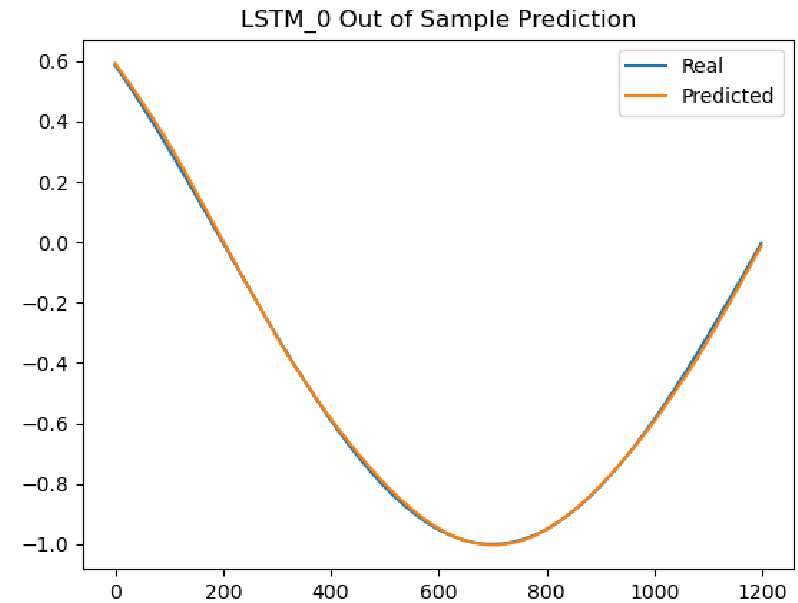

Figure 7: LSTM Out of Sample Prediction

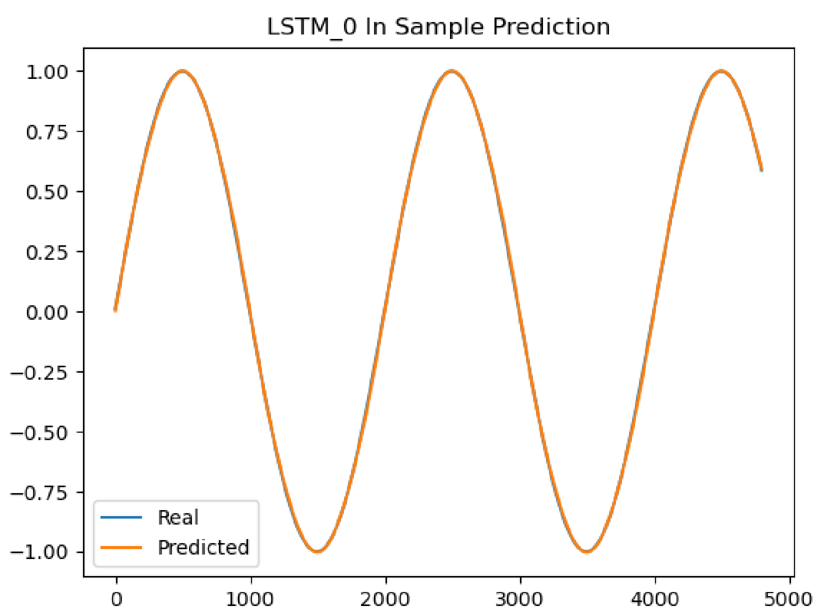

Figure 8: LSTM In Sample Prediction

R2	0.999
MAE	0.008
MSE	0.00
MAPE	0.048

Table 4: LSTM Accuracy Metrics

4. A stochastic series sampled from a sinusoidal wave with period 20 seconds, with a sample rate of 100 Hz, plus random white noise i.e., random variable with zero mean and 1 variance. Control the amplitude of the noise with a fractional number, e.g. 0.1. Design an RNN and a LSTM for two-step prediction. The output vector has a size of 2. Set the input vector size by yourself.

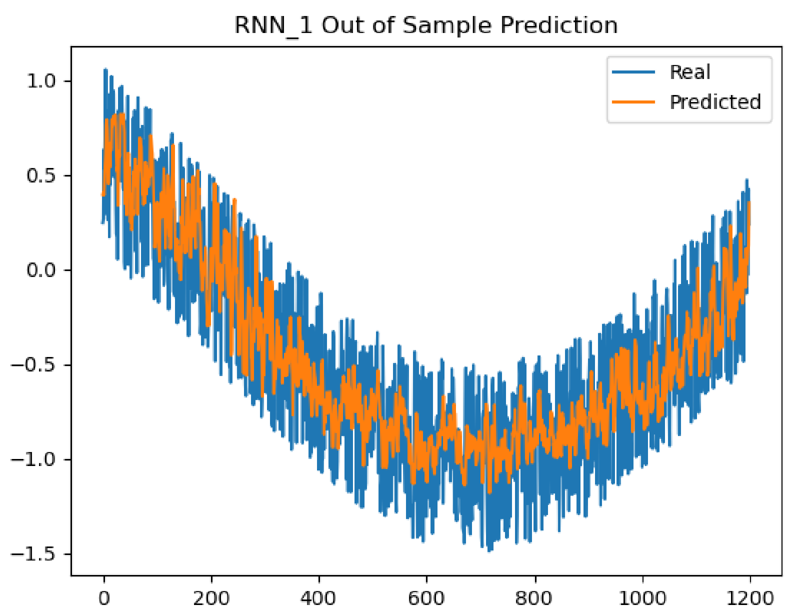

Figure 9: RNN Out of Sample Prediction

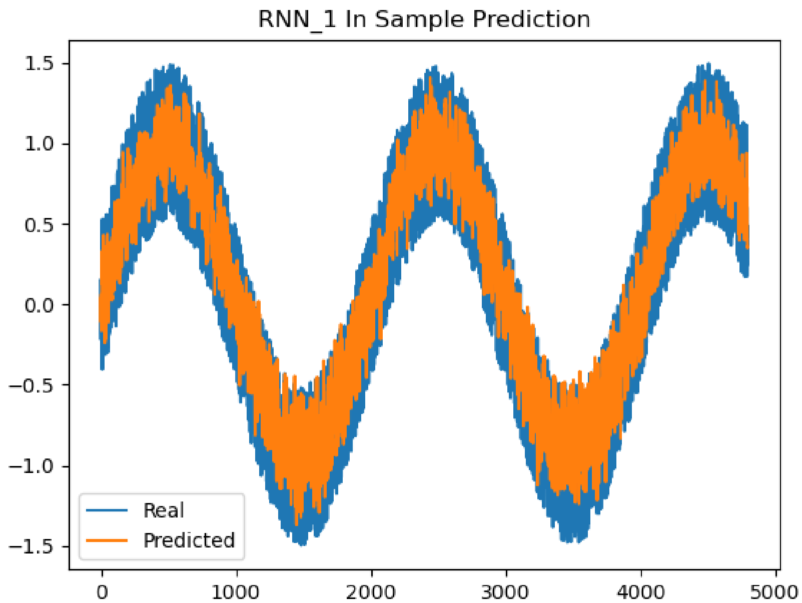

Figure 10: RNN In Sample Prediction

R2	0.334
MAE	0.275
MSE	0.107
MAPE	1.961

Table 5: RNN Accuracy Metrics

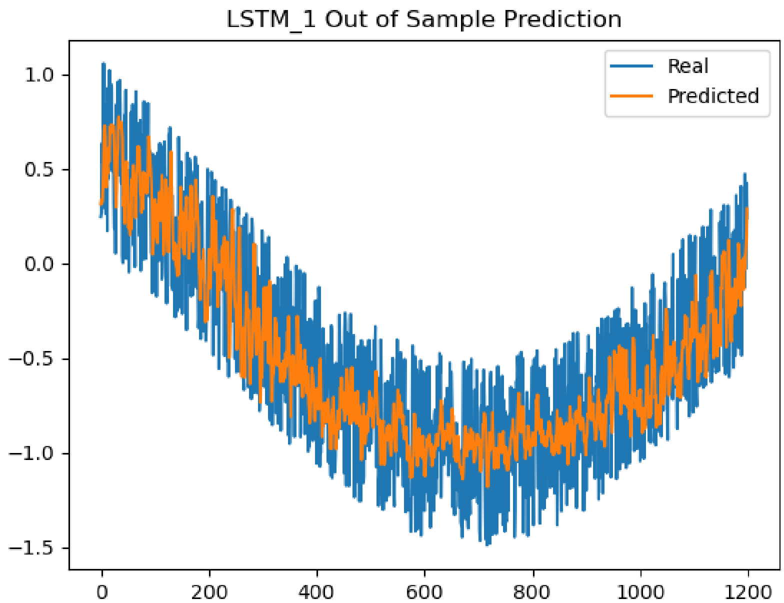

Figure 11: LSTM Out of Sample Prediction

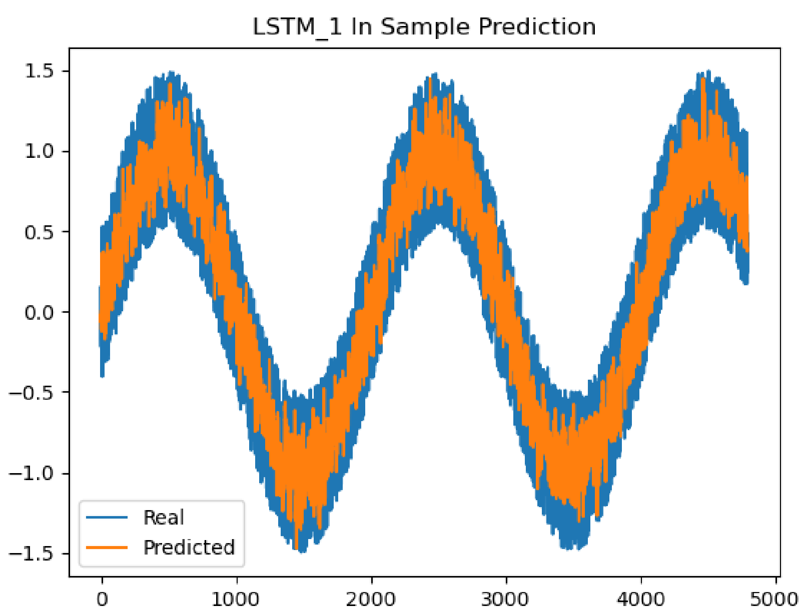

Figure 12: LSTM In Sample Prediction

R2	0.503
MAE	0.266
MSE	0.099
MAPE	1.131

Table 6: LSTM Accuracy Metrics

Discussion:

1. How have you designed the neural network for each series? What hyper-parameters do you use in each case?

   All Models use the "Adam" optimizer with a "MSE" Loss function. 100 Epochs were chosen based on trial and error.

2. Can the neural network fit well to the specific series well? What are the accuracy merits?

   Yes. Using accuracy metric such as R2, MSE, MAE and MAPE, once can check if the model accuracy meets their specific requirements. White noise is inherently unpredictable and thus predictions by any model will not be very accurate.

3. How is the performance of LSTM in comparison with RNN? Is the LSTM outperforming the RNN in general?

   LSTM tends to outperform the RNN generally however in my models, there is no significant difference in model prediction accuracies.

1.2 Predict white noise, random walk, an ARMA process using neural networks

We use three synthetic data sets, each with 1000 data points.

1. A pure white-noise signal.
2. A random-walk series.
3. A stationary series generated by an ARMA(2, 2) process. Make sure that the process with right parameters generates a stationary series.

Your task is to design and test an neural network to build a prediction model for the three series, ie., white noise, random-walk, and ARMA(2, 2) process

An MLP, RNN, LSTM and ARIMA model was used to model each type of data.

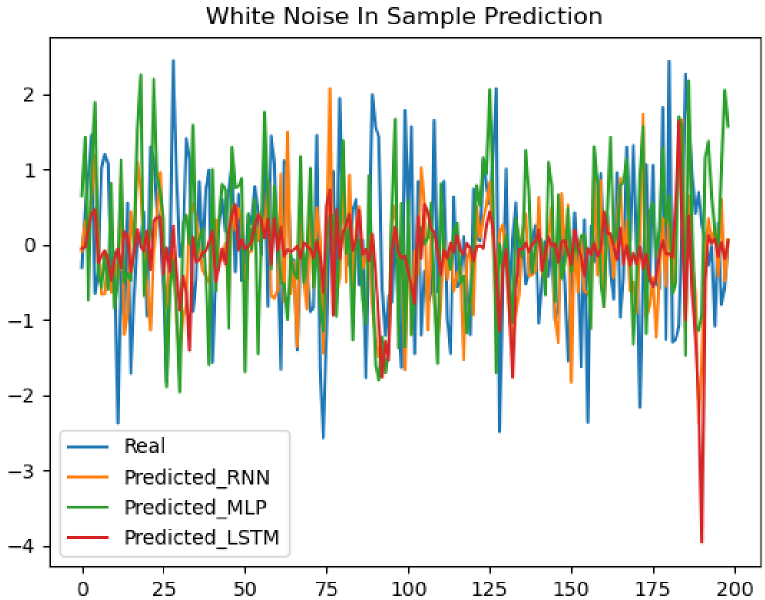

Figure 13: White Noise In Sample Prediction

	R2	MAE	MSE	MAPE
MLP	-1.32	1.070	1.737	5.78
RNN	-2.138	0.936	1.456	14.8
LSTM	-4.175	0.865	1.228	24.29

Table 7: White Noise Series Accuracy Metrics

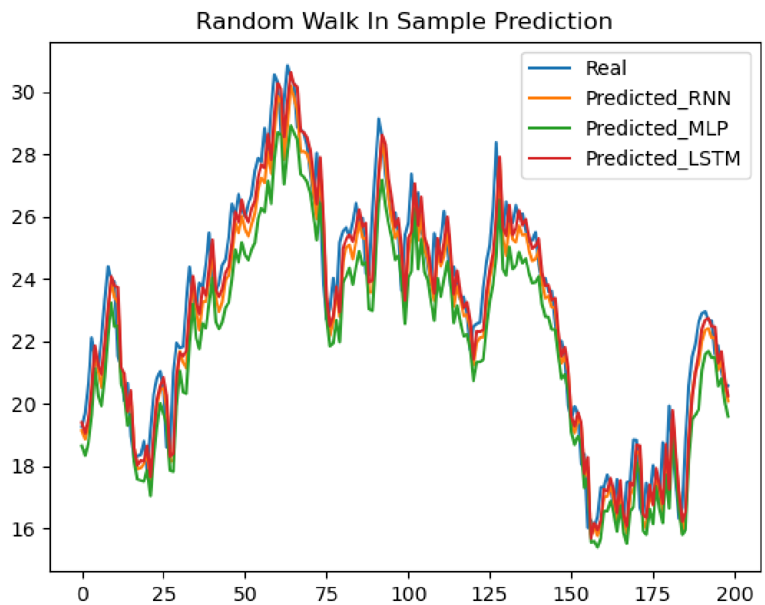

Figure 14: Random Walker In Sample Prediction

	R2	MAE	MSE	MAPE
MLP	0.923	0.821	1.045	0.036
RNN	0.920	0.850	1.088	0.037
LSTM	0.927	0.805	0.998	0.035

Table 8: Random Walker Series Accuracy Metrics

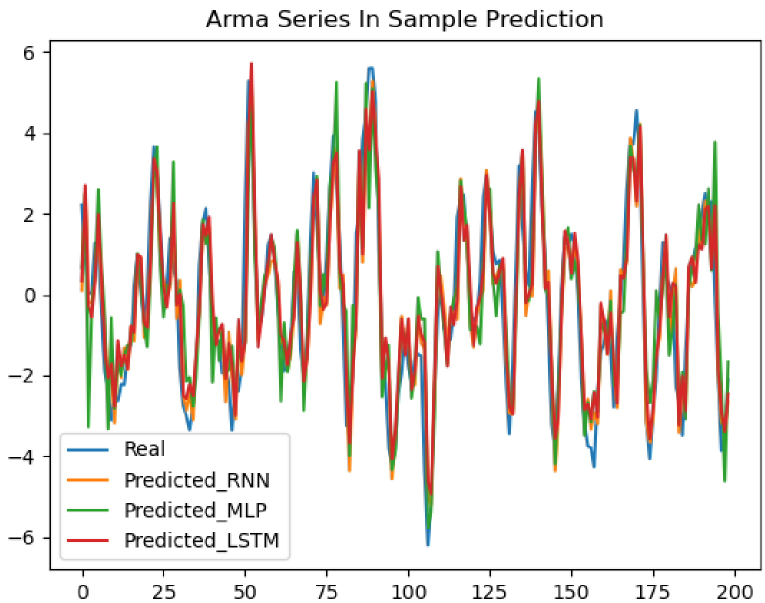

Figure 15: ARMA In Sample Prediction

	R2	MAE	MSE	MAPE
MLP	0.662	0.982	1.489	3.148
RNN	0.744	0.813	1.097	0.711
LSTM	0.755	0.795	1.037	11.93

Table 9: ARMA Series Accuracy Metrics

Based on the accuracy metrics of each model in each series, the most suitable model was determined to be an LSTM, however only by a small margin. Effectively, either mode could be used.

Discussion:

1. How have you designed the neural network? What hyper-parameters do you choose? Give short motivation.

   The models use an input vector of 5. Loss function is "MSE" and optimizer is "Adam". Activation function is "Relu". Epochs are set at 100.

   Parameters were chosen based on visual inspection of different input vectors. Optimizer, activation function and loss function are ketpt consistent across the project for better evaluation.

2. Can the neural network fit well to the white noise series?

   Not really, since white noise is stationary but random and unpredictable.

3. Can the neural network fit well to the random walk series?

   All neural networks seem to fit the random walker series as indicated by the MSE.

4. Can the neural network fit well to the ARMA process? Why or Why not?

   Yes however, MSE indicated that MLP performs the worst.

1.3 Task 1.3 Comparison with ARIMA-based modeling and prediction

Generate a certain-length (e.g. 50 points) Fibonacci series and add standard Gaussian noise by yourself. Control the signal-noise ratio properly by controlling the amplitude of the signal and the noise, e.g. 95% of signal, 5% of noise

Your task is to build four models for the Fibonacci series, use the models to predict future values, and make a comparative evaluation.

1. Generate a Fibonacci series. You decide the length of the series to generate. Split the data into a training set and a test set. You decide the splitting ratio.

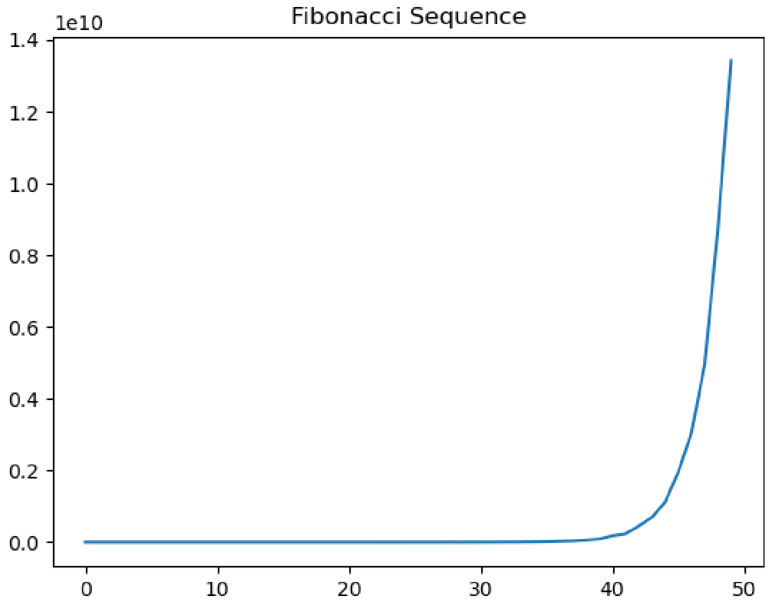

Figure 16: Fibonacci Sequence

2. Build an MLP model for the series and use it for prediction.

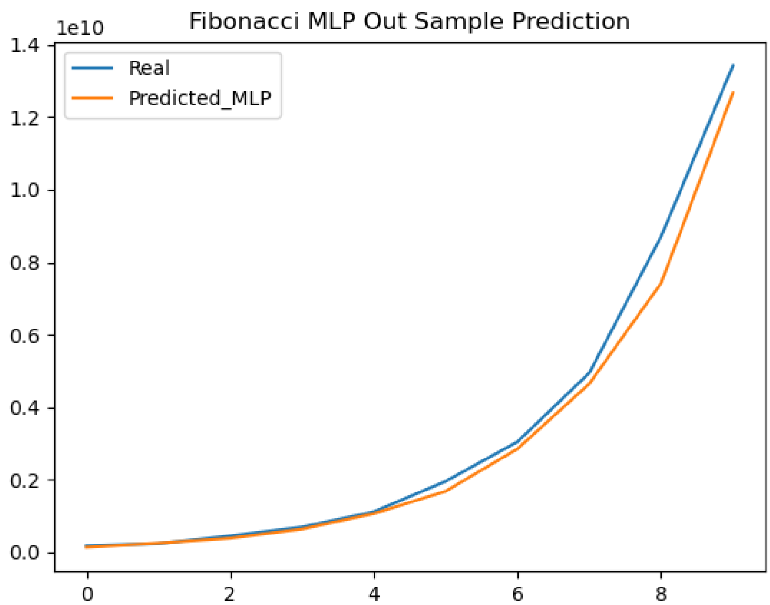

Figure 17: Fibonacci Sequence MLP Out of Sample Prediction

R2	0.992
MAE	191 e+10
MSE	128 e+15
MAPE	0.101

Table 10: MLP Accuracy Metrics

3. Build an RNN model for the series and use it for prediction.

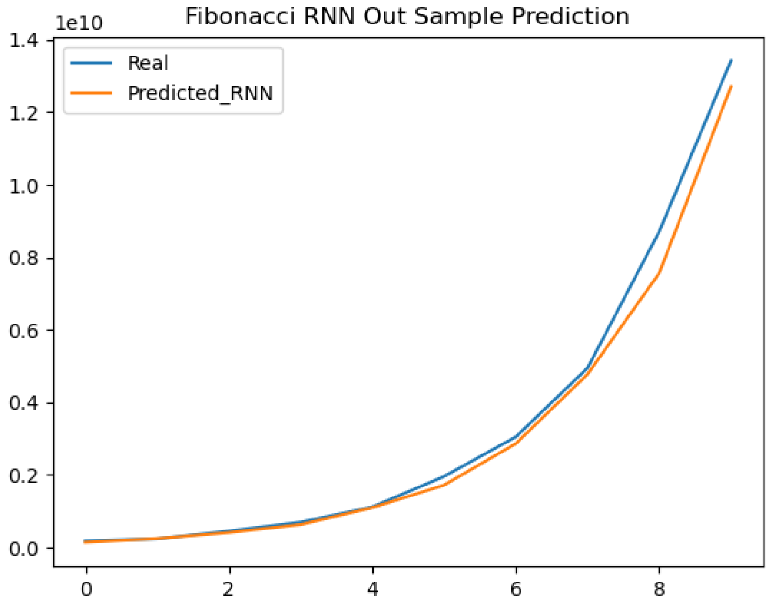

Figure 18: Fibonacci Sequence RNN Out of Sample Prediction

R2	0.984
MAE	288 e+10
MSE	228 e+15
MAPE	0.102

Table 11: RNN Accuracy Metrics

4. Build an LSTM model for the series and use it for prediction.

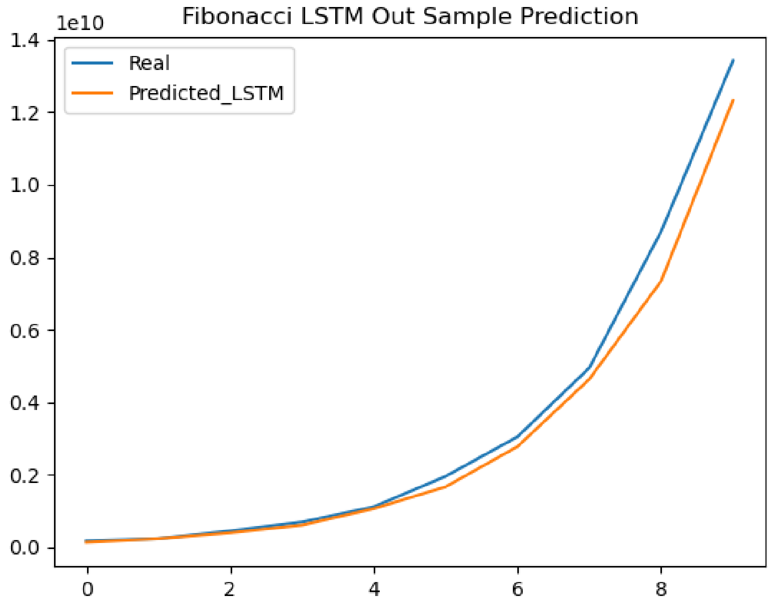

Figure 19: Fibonacci Sequence LSTM Out of Sample Prediction

R2	0.986
MAE	272 e+10
MSE	204 e+15
MAPE	0.105

Table 12: LSTM Accuracy Metrics

5. Build an ARIMA model for the series and use it for prediction

   ADF Test: p value of 0.994 means the data is not stationary. Differencing was employed.

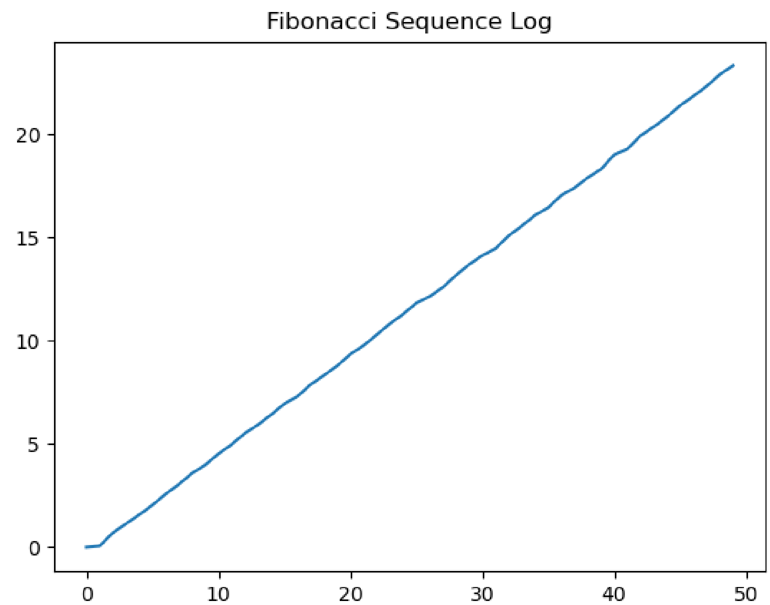

Figure 20: Fibonacci Sequence Log Transformed

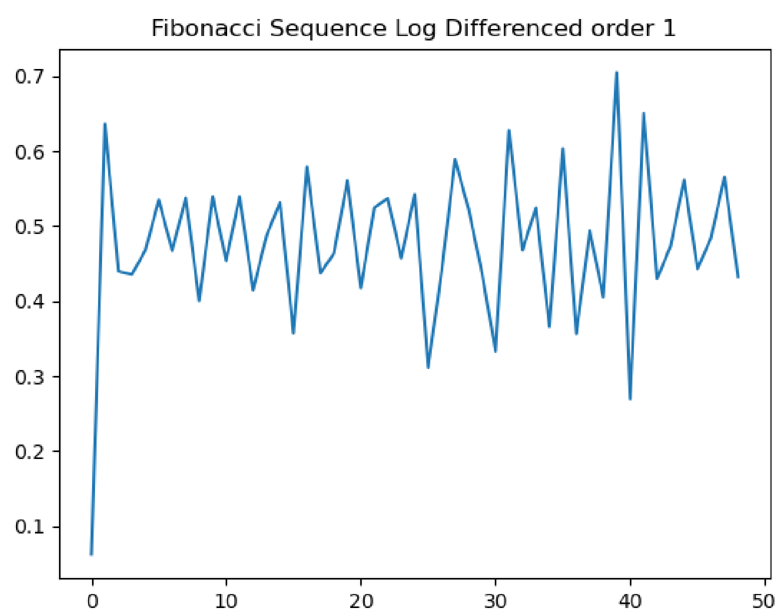

Figure 21: Fibonacci Sequence Log Transformed - Difference Order 1

ADF Test: p value of 3.86e-04 means the data is stationary as it is less than 0.05.

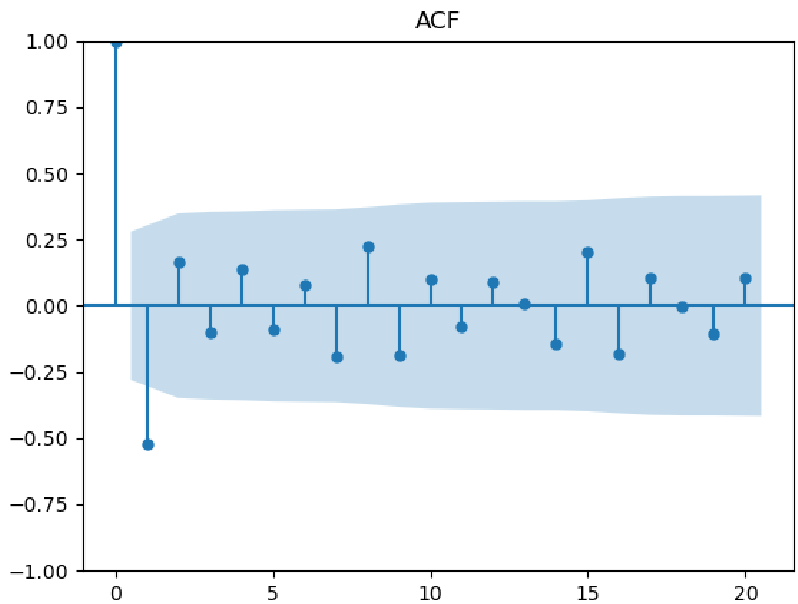

Figure 22: Fibonacci Sequence ACF

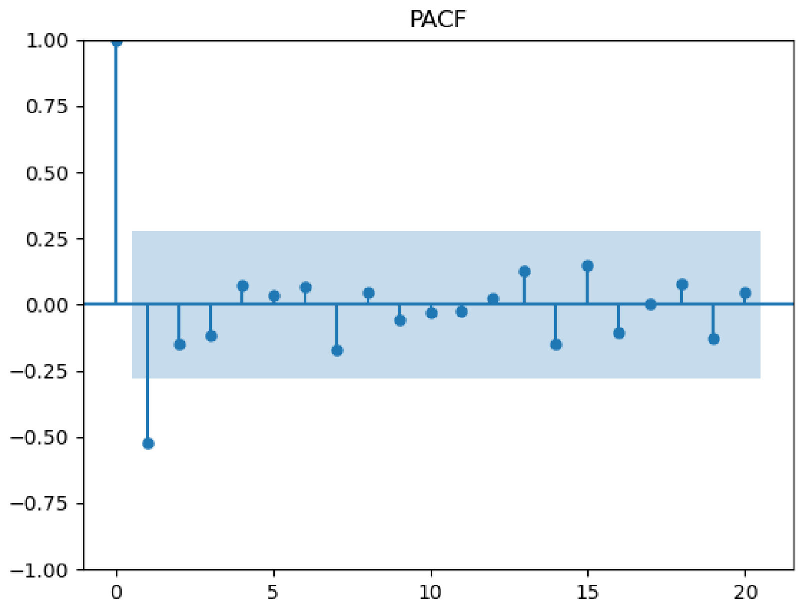

Figure 23: Fibonacci Sequence PACF

Based on the PACF and ACF graph, a p and q value of 2 was chosen. Given that a stationary series was achieved after first order differencing, a d value of 1 was chosen for the ARIMA model

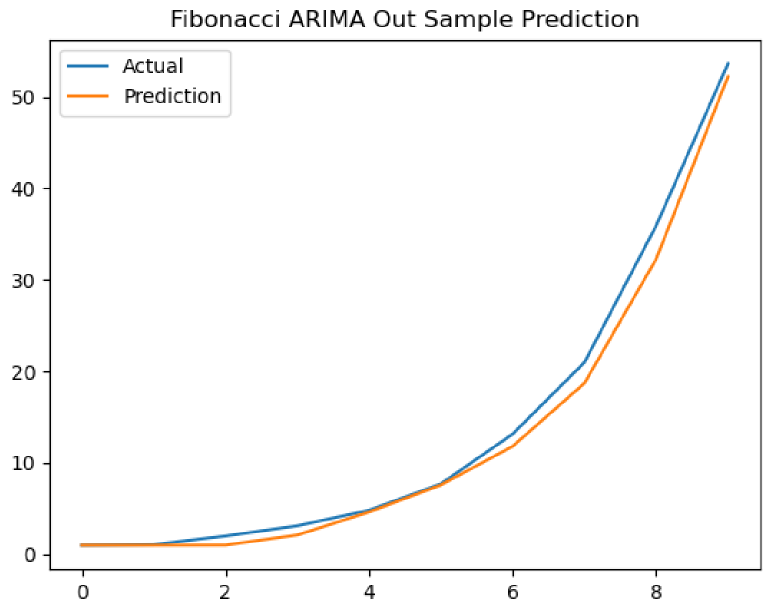

Figure 24: Fibonacci Sequence ARIMA Prediction

R2	0.966
MAE	0.154
MSE	0.066
MAPE	0.17

Table 13: ARIMA Accuracy Metrics

6. Compare the accuracy (MSE, MAE, MAPE of the prediction errors) of the four methods for the training set and the test set.

	R2	MAE	MSE	MAPE
MLP	0.992	1.91e10	1.28e15	0.102
RNN	0.984	2.88e10	2.28	0.102
LSTM	0.986	2.72e10	2.04	0.105
ARIMA	0.966	0.154	0.06	0.17

Table 14: Fibonacci Sequence Accuracy Metrics

Discussion:

1. How have you designed the neural networks? What hyper-parameters do you choose? Give short motivation.

   The models use an input vector of 2. Loss function is "MSE" and optimizer is "Adam". Activation function is "Relu". Epochs are set at 100.

   Parameters were chosen based on visual inspection of different input vectors. Optimizer, activation function and loss function are kept consistent across the project for better evaluation.

2. How have you trained your neural networks? Report the training epoch, learning rate, optimization algorithm.

   I split the Fibonacci sequence into training (80%) and testing(20%) data. Models were trained on the training data set and their predictions were compared with the testing data set.

3. Can your MLP, RNN, LSTM networks fit well to the Fibonacci series? Which one is best?

   In my case, it seems the MLP fit the best. However, MSE indicates they are not very good at fitting this sequence.

4. Can your ARIMA model fit well to the Fibonacci series?

   Though all can fit, ARIMA is the best if we use the MSE as the metric.

5. Which modeling approach, neural network based or ARIMA based, gives a better performance? Why? Discuss the pros and cons of different modeling approaches

   ARIMA seems to give better performance. This might be a result of non-ideal hyper-parameters set for the neural networks.

   Neural Networks are powerful and flexible, capable of modeling complex, non-linear relationships in large datasets, but they require significant data, computational resources, and careful tuning to avoid overfitting.

   ARIMA Models are simpler, more interpretable, and effective with smaller datasets, but they assume linearity and stationarity, which may limit their applicability to more complex time series data.

6. Which model, ARIMA or NN, is more tolerant to noise? Increasing the noise ratio and report how the accuracy of the two models will be worsened.

   It seems Neural networks are more tolerant to noise.

2 Decomposition-based anomaly detection

2.1 Anomaly identification in global land temperature changes

Your task is to (1) insert one or two anomaly points as ground truth, and then (2) identify the anomaly points from the above data set for the 'LandAverageTemperature' data using the decomposition method

An anomaly was placed 1995-01-01.

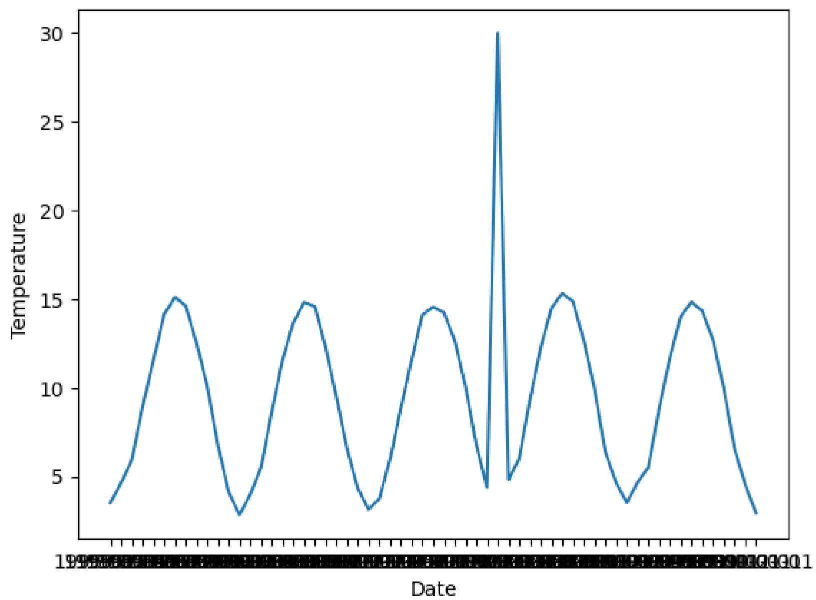

Figure 25: Global Temperature with Anomaly

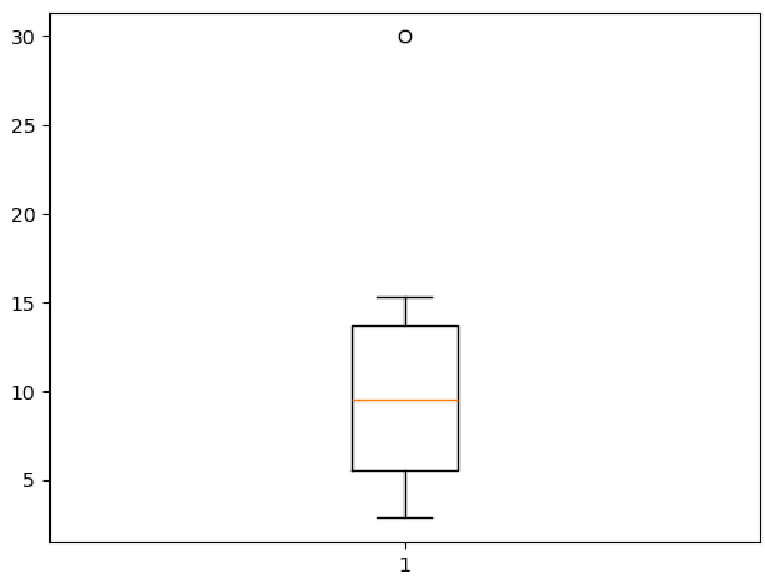

Figure 26: Global Temperature Anomaly Box Plot

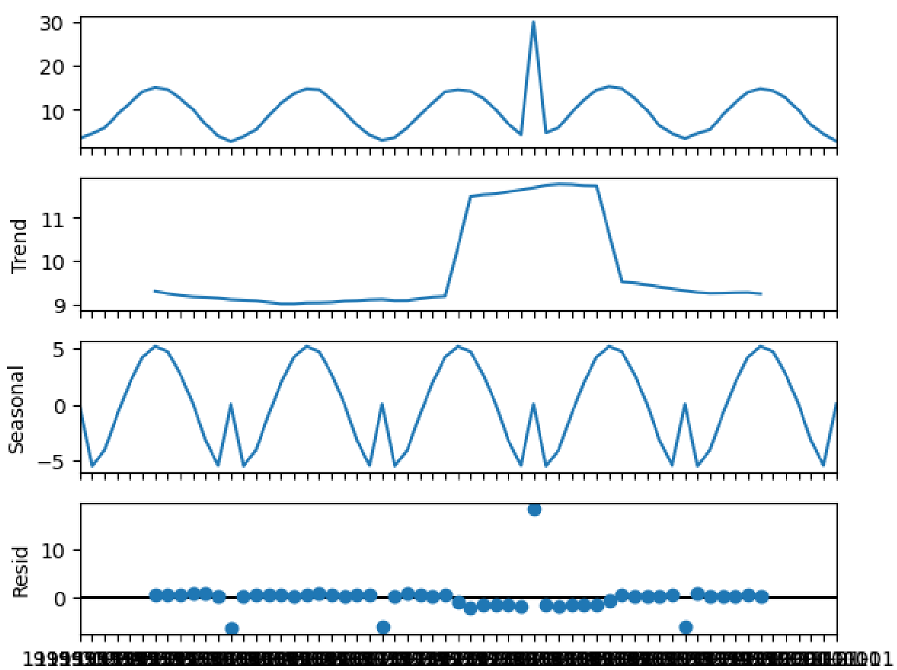

Figure 28: Global Temperature Seasonal Decomposition

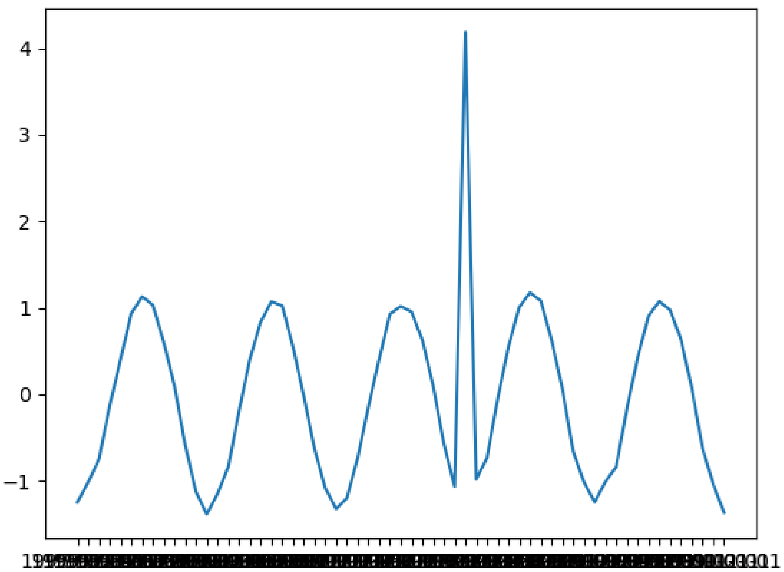

Figure 27: Global Temperature Anomaly Zscore Plot

Discussion:

1. Can the decomposition clearly separate the trend, season (constant period), and remainder components

   The decomposition can separate the season and remainder components however the data set appears to be too small to separate trend lines. It would requires to use the entire data set to extract the trend line.

2. When decomposing the series, is there a general rule to determine which part belongs to a trend, a season, or a remainder? Or is it embedded in and thus dependent on each individual algorithm?

   No general rule, it depends on the individual algorithms. However each components has specific characteristics to look out for. Seasonality is periodic, trend lines show a pattern over long time periods etc.

3. Is there a growing tendency in the trend series?

   Yes, temperature seems to be steadily growing.

3 Prediction-based anomaly detection

3.1 Anomaly detection by prediction

Your task is to identify the anomaly points from the Global Land Temperature Anomaly data set using the prediction-based anomaly detection with ARIMA. Suppose that the anomaly ratio is 2%

1. Exploratory data analysis. Draw seven graphs: line plot, histogram, density plot, heatmap, box plot, lag-1 plot, and lag-2 plot for the series. If the series is not stationary, differencing the series, until it is stationary, but not over-differenced.

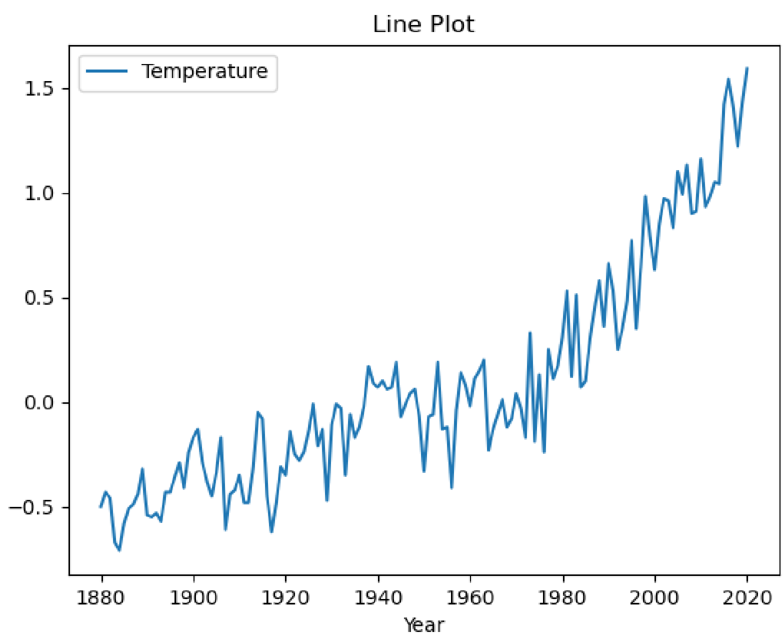

Figure 29: Global Temperature Line Plot

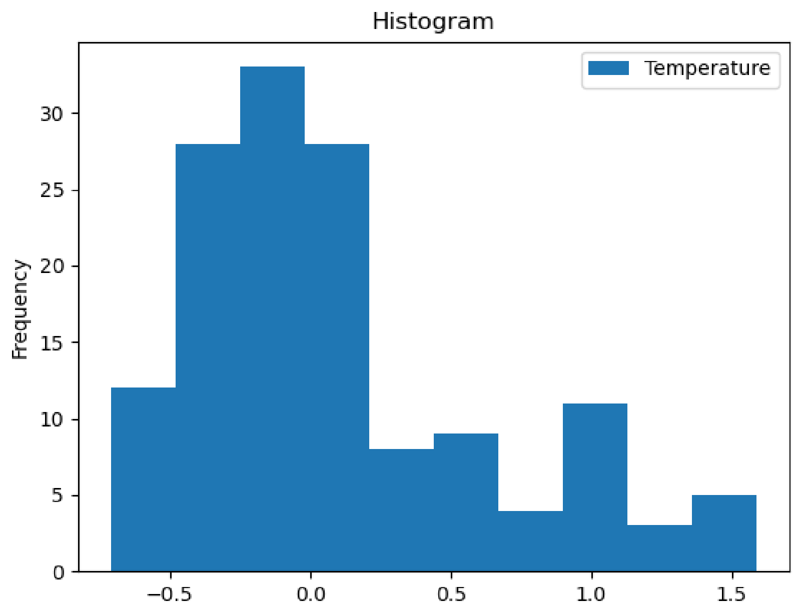

Figure 30: Global Temperature Histogram

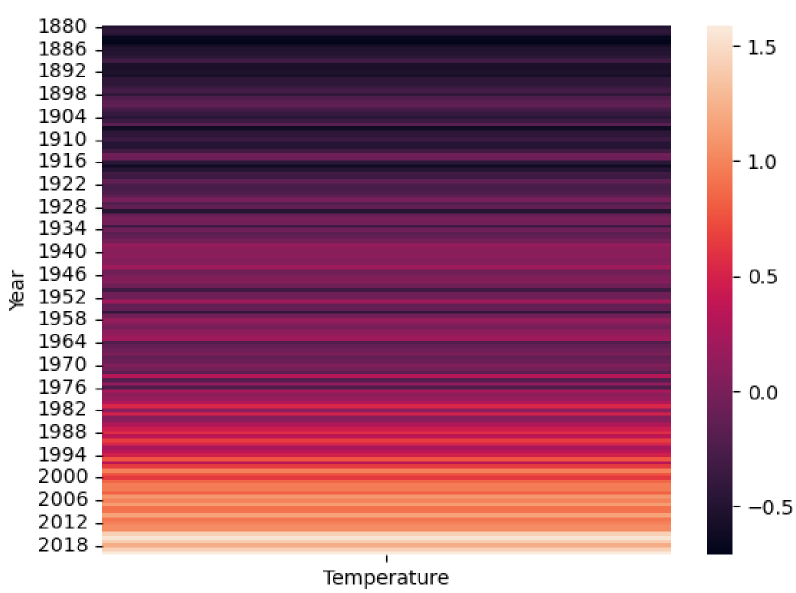

Figure 32: Global Temperature Heat Map

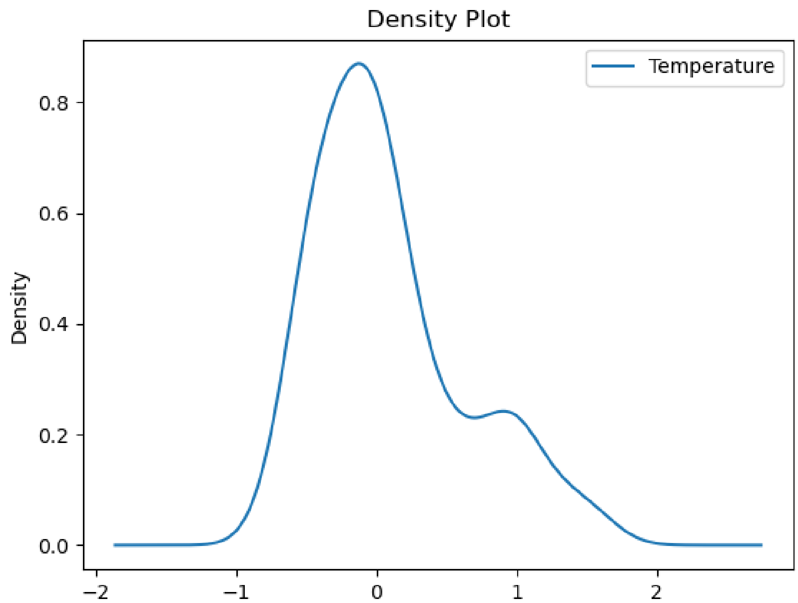

Figure 31: Global Temperature Density Plot

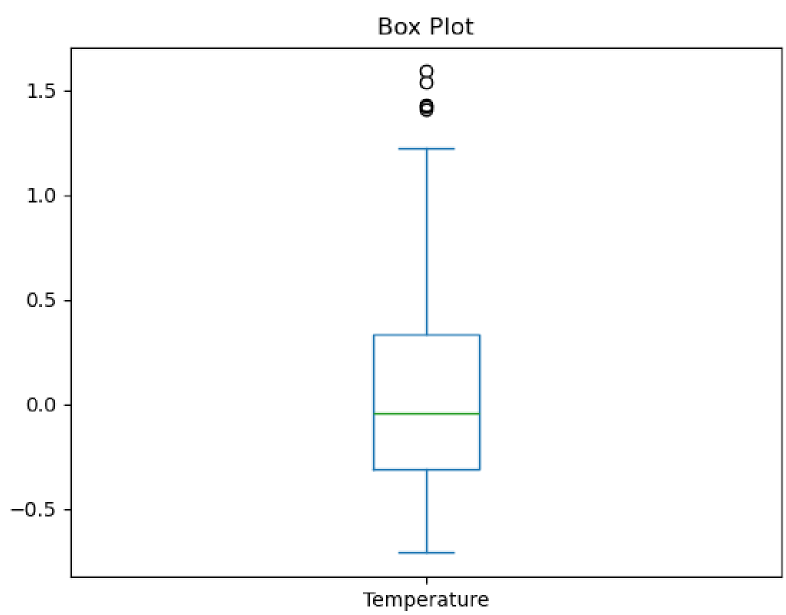

Figure 33: Global Temperature Box Plot

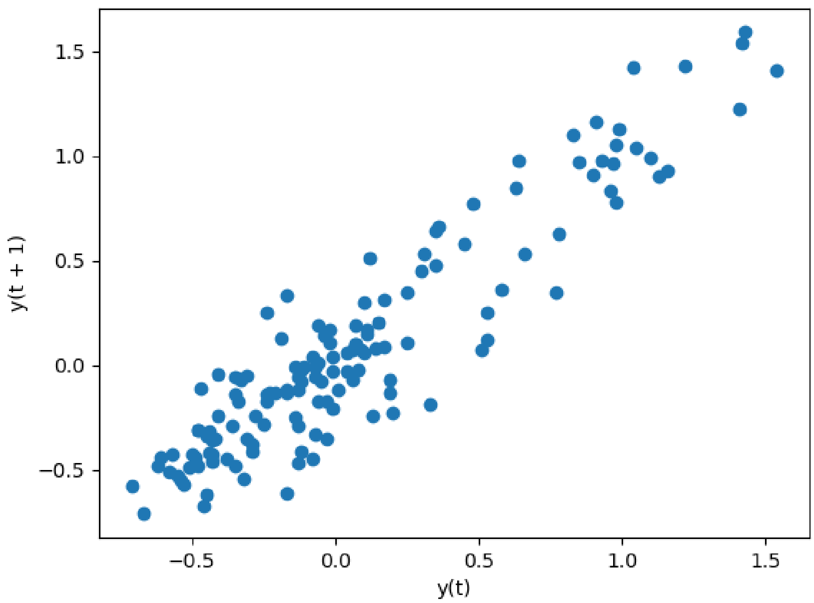

Figure 34: Global Temperature Lag-1 Plot

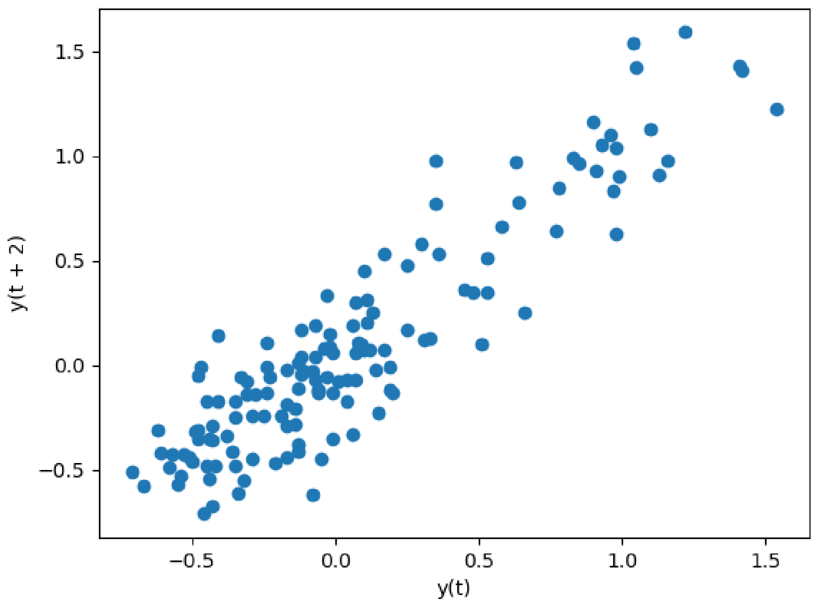

Figure 35: Global Temperature Lag-2 Plot

ADF Test: p value of 0.993 indicates data is not stationary. Differencing in employed.

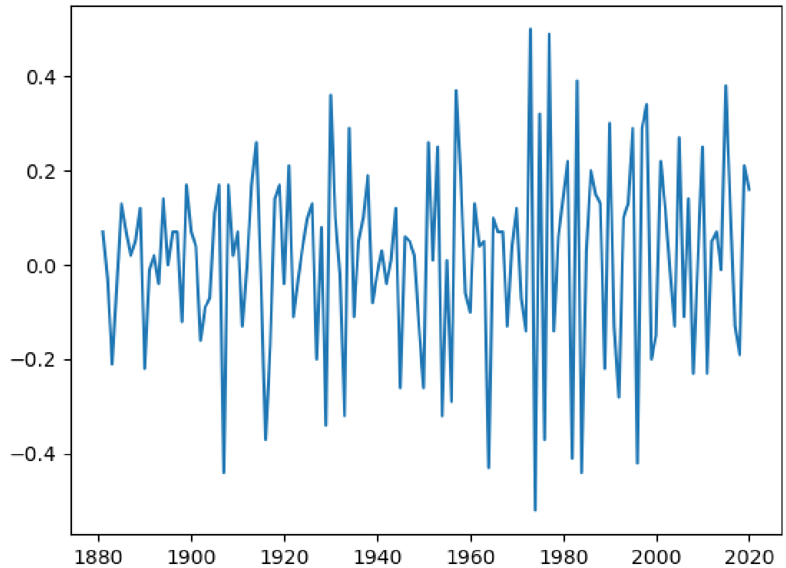

Figure 36: Global Temperature Differenced 1 Plot

ADF Test: p value of 0.0 indicates data is stationary as value is less than 0.05

2. Feature extraction. Summarize the statistical features of the series, such as mean, standard deviation. Plot the temporal correlation of the series by drawing its acf and pacf graphs.

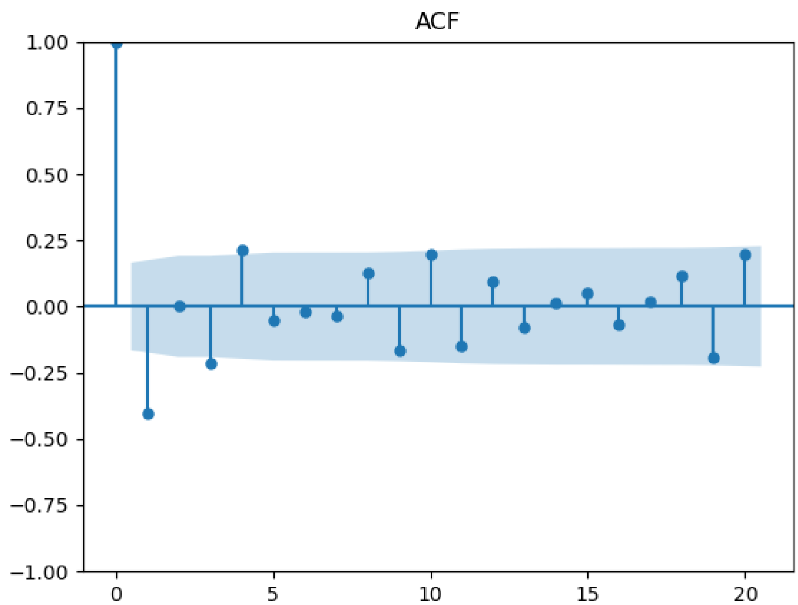

Figure 37: Global Temperature ACF

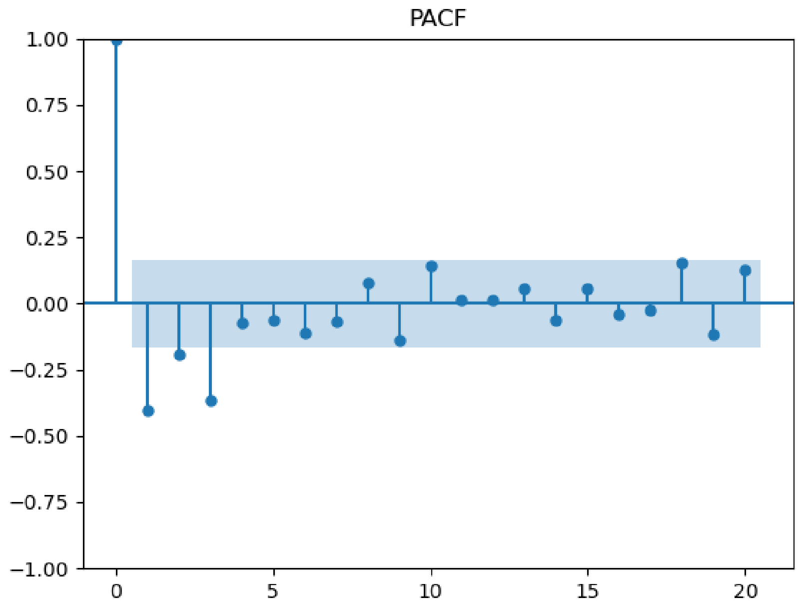

Figure 38: Global Temperature PACF

Mean	0.014
Standard Dev	0.199
Variance	0.039

Table 15: Series Feature Statistics

3. Model construction and selection. Follow the Box-Jenkins methodology step by step to construct an ARIMA(p, d, q) process to model the series.

   p and q values of 2 and 4 were chosen based on ACF and PACF graphs. d value of 1 was chosen because of order-1 differencing

4. Prediction. Use the model to do in-sample prediction, and generate the prediction error series. Check if the remainder (prediction error) series is random

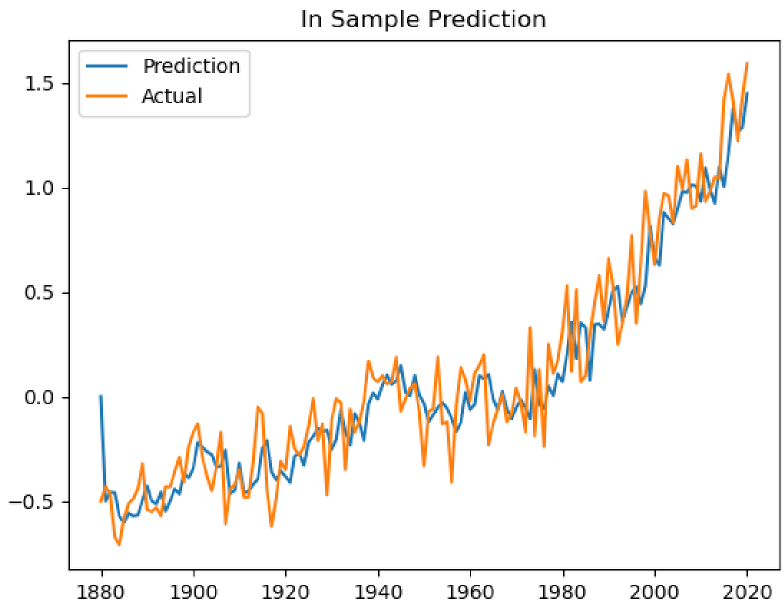

Figure 39: Global Temperature ARIMA Model Prediction

Ljung-box test: p value of 0.25 indicates the residual data is random.

5. Anomaly definition and detection. Implement the Z-score, Boxplot criterion to identify the outliers and mark the anomaly points in the original series.

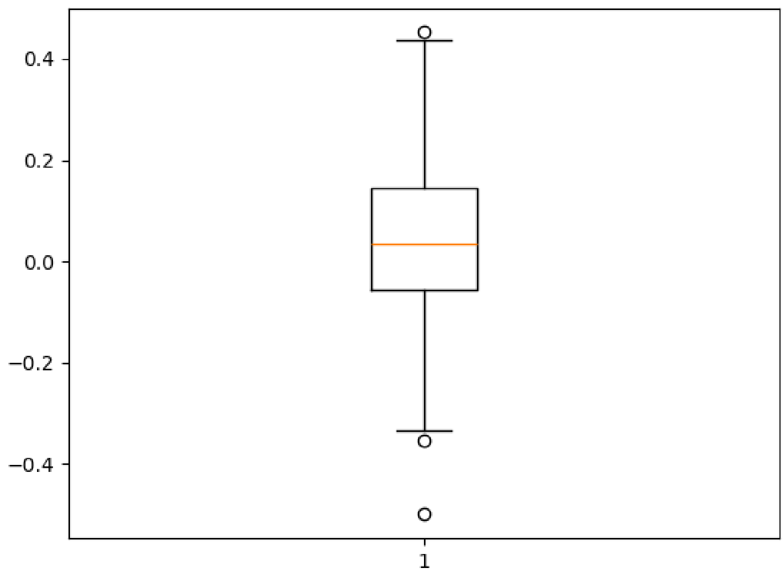

Figure 40: Global Temperature Series Residuals Boxplot

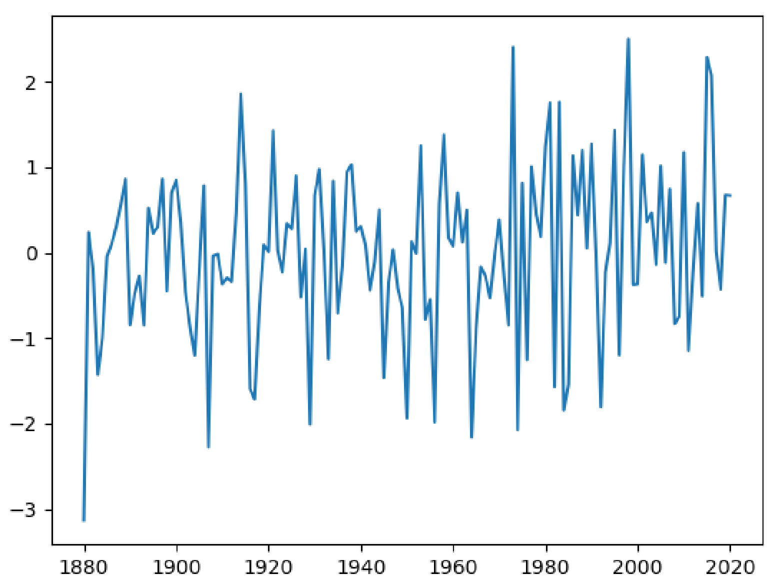

Figure 41: Global Temperature Series Residuals Zscore

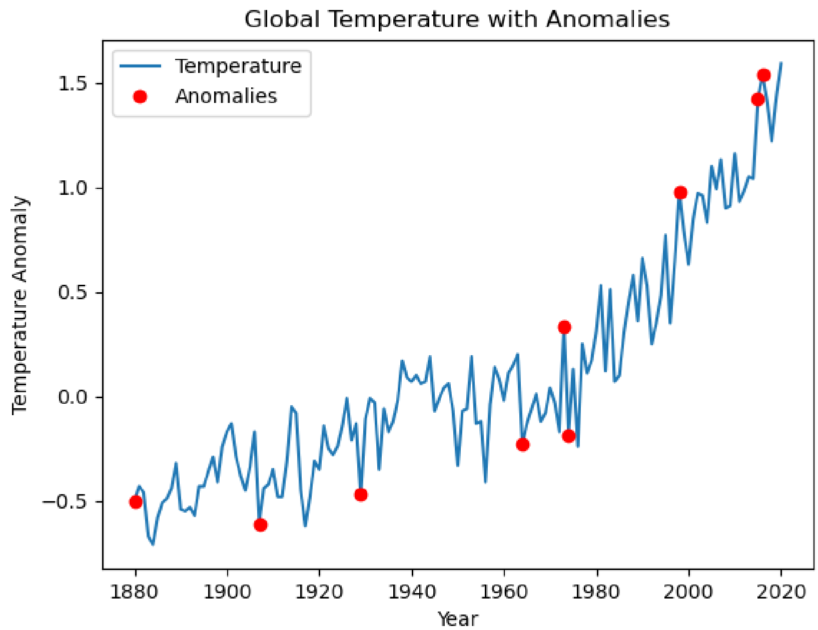

Figure 42: Global Temperature Series with Anomalies

Discussion:

1. Since this task uses a different approach (prediction-based anomaly detection) from Task 2 which uses decomposition for anomaly detection, describe what the differences of the two methods are?

   Seasonal decomposition extracts features of the data set including trend lines and seasonality. By analyzing these characteristics across the data set, anomalies that violet the pattern can be identified.

   Prediction trains a model based on a sub set of the data set. The model is used to predict the remainder of the data set. The actual and remainder data sets are compared and outliers identified.

2. Do they achieve the same results? Why or Why not?

   No. It depends on various factors such as patterns in the data set, the accuracy of the model etc

3. Given the anomaly ratio of 2%, what is the value of z-score?

   Zscore threshold is 2

3.2 Anomaly detection in ECG signals with LSTM

We use the ECG signals from the MIT-BIH Arrhythmia database. Your task is to build a prediction model for the ECG series using LSTM

1. Exploratory data analysis. Draw the line plot, lag-1 plot, and lag-2 plot for the two signals in the data set

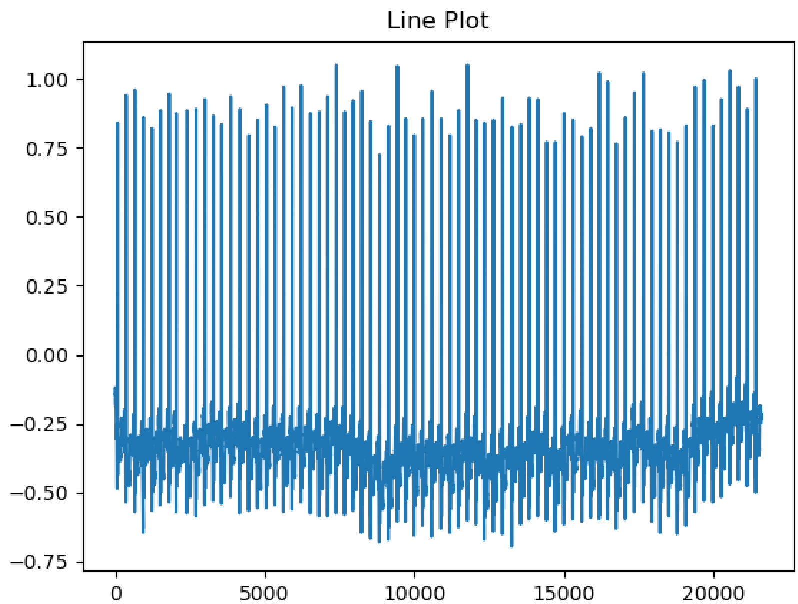

Figure 43: ECG MLII Line PLot

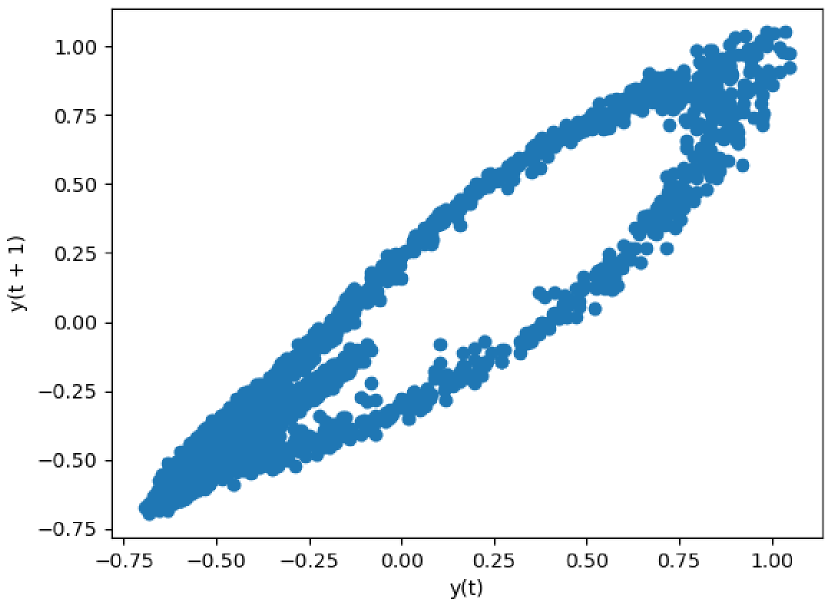

Figure 44: ECG MLII Lag-1 PLot

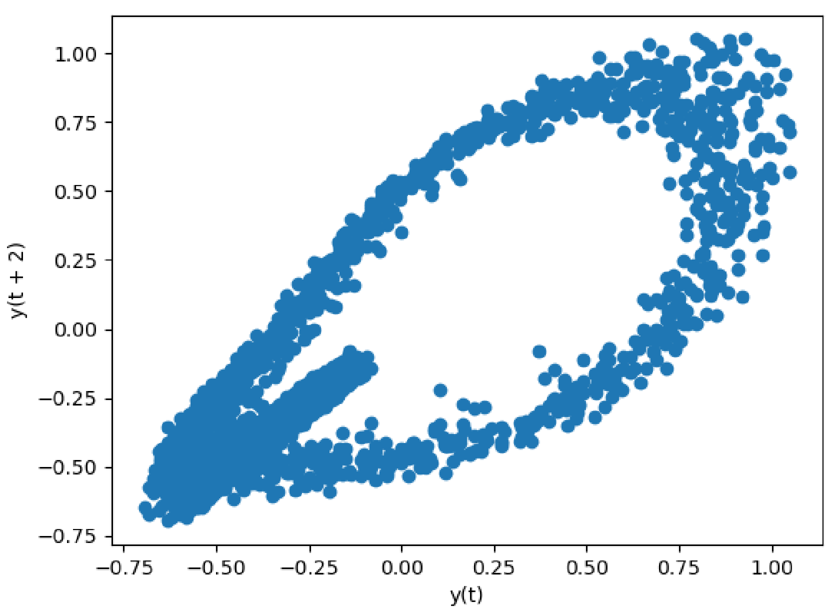

Figure 45: ECG MLII Lag-2 PLot

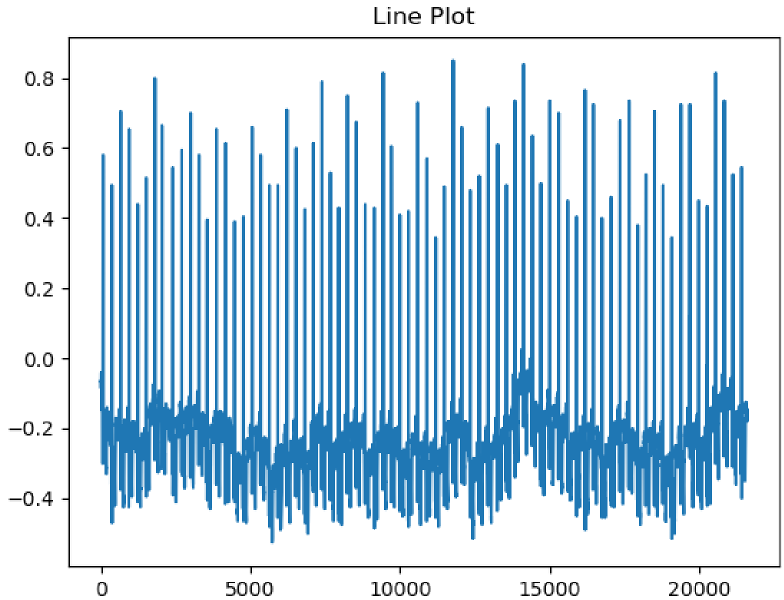

Figure 46: V5 MLII Line PLot

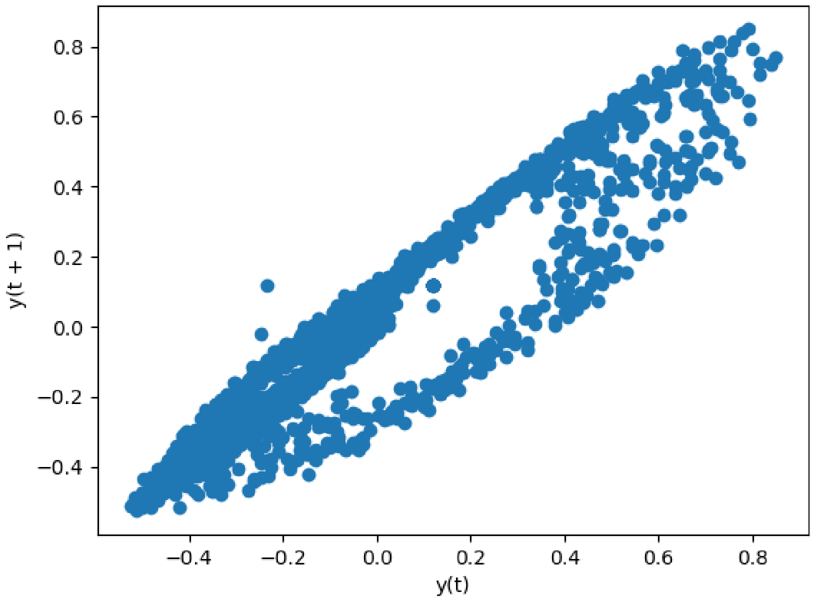

Figure 47: ECG V5 Lag-1 PLot

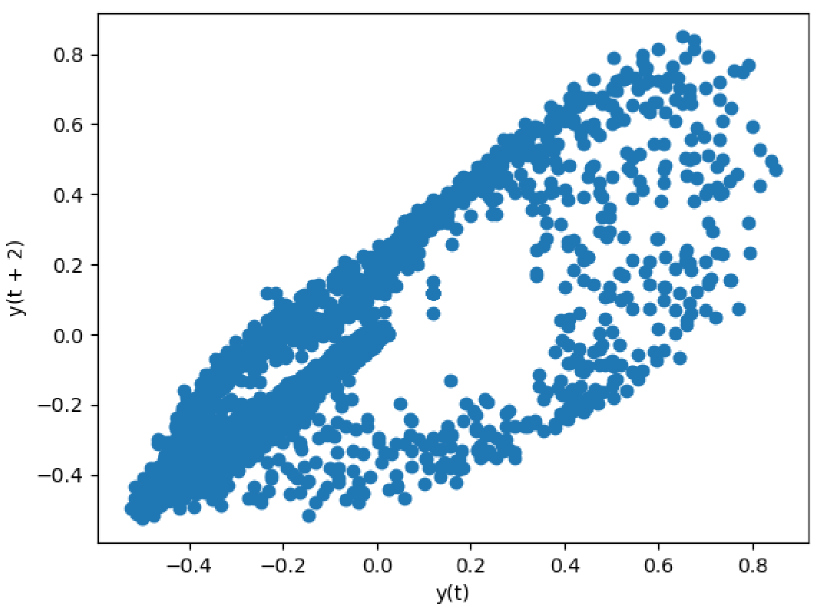

Figure 48: ECG V5 Lag-2 PLot

2. Model construction. Split the data set into a training set (80%) and a test set (20%). Design and train an LSTM which can predict the next value given a few known values

    (a) When you re-organize the data structure, let the input vector be a size of 4, 8, 16 (Each time, use one of the 3 options) and the output vector be a size of 1.

    (b) Treat the time-series data in two ways: 2 individual uni-variate series, 1 bi-variate series

3. Validate the quality of the model. Calculate the MSE, MAPE etc. of the prediction model for the input size being 4, 8 and 16.

	n=4	n=8	n=16
MLII	0.00019	0.00015	0.00015
V5	0.00030	0.00021	0.00021
MLII - V5	0.00024	0.00018	0.00017

Table 16: Univariate / Bivariate Model MSE

	n=4	n=8	n=16
MLII	0.06194	0.03260	0.03366
V5	0.05596	0.07135	0.05265
MLII - V5	0.08353	0.05834	0.05849

Table 17: Univariate / Bivariate Model MAPE

	n=4	n=8	n=16
MLII	0.01024	0.00902	0.00935
V5	0.01070	0.00881	0.00889
MLII - V5	0.01024	0.00894	0.00852

Table 18: Univariate / Bivariate Model MAE

4. Anomaly definition and detection. Calculate the prediction error series. Assume 0.5% of error rate, find the anomaly points.

    MLII

    N=4 : [67 73 79 376 656 933 1501 1507 1517 1789 1801 2102 2413 2416 2698 2992 3267 3274 3550 3558 3840 3843]

    N=8 : [66 375 378 932 1500 1506 1788 2100 2104 2409 2412 2416 2993 3264 3274 3278 3503 3548 3549 3851 4133 4145]

    N=16 : [65 373 374 1503 1505 1515 1784 1799 2095 2395 2408 2411 2981 2992 3265 3277 3548 3549 3838 3841 4132 4144]

V5 N=4 : [67 75 365 372 374 375 376 665 671 1222 2392 2393 2395 2404 2409 2410 2952 2953 2963 2986 3547 3552]

N=8 : [371 373 664 943 1221 1512 1513 1516 1798 2391 2392 2403 2406 2408 2409 2951 2952 2962 3267 3270 3847 4139]

N=16 : [72 663 936 1230 1512 1515 2390 2391 2402 2403 2405 2410 2703 2706 2950 2951 2961 2962 2969 2986 3270 3554]

3265 3277 3548 3549 3838 3841 4132 4144]

MLII - V5 N=4 : [293 592 600 1160 1172 1173 1452 1748 2040 2636 2929 3211 3219 3784 4387 4684 4692 4985 5542 5552 5555 5834 5837 6118 6712 6713 6715 6723 6724 6725 6729 6730 7018 7272 7273 7283 7284 7309 7583 7595 7867 7878 8160 8452]

N=8 : [301 591 599 1159 1160 1172 1738 1747 2337 2644 3218 3221 3501 3783 3791 4984 5541 5551 5832 5833 6118 6711 6712 6713 6714 6723 6726 6729 6731 7017 7027 7271 7272 7282 7283 7299 7300 7305 7307 7582 7595 8158 8159 8451]

N=16 : [597 598 880 1158 1159 1160 1167 1175 1450 1451 1737 1746 1750 2346 2643 3217 3496 4395 4684 4983 5545 5550 5831 5832 5835 6107 6117 6710 6711 6713 6721 6722 6725 6726 7023 7026 7270 7271 7281 7282 7284 7306 7590 8165]

5. Visualize the anomaly points in the prediction error series.

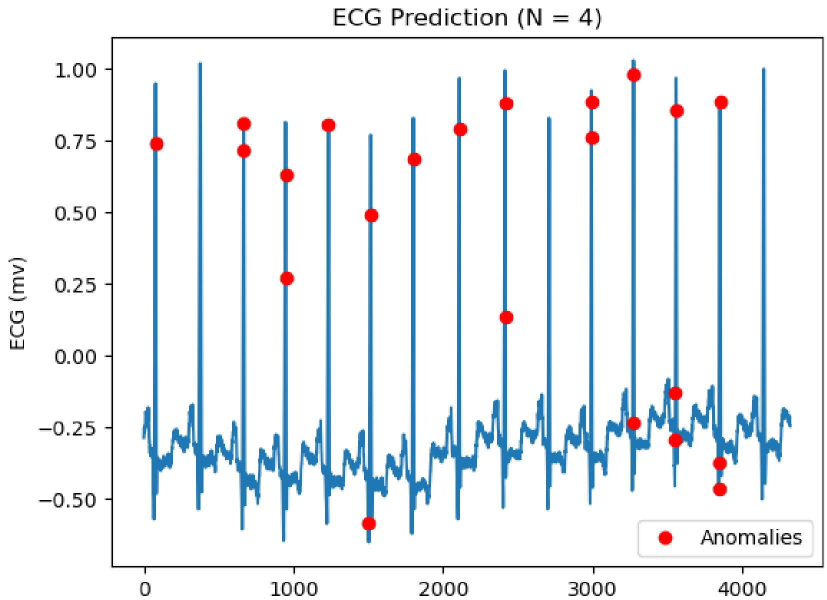

Figure 49: LSTM MLII Prediction W/ Anomaly (N=4)

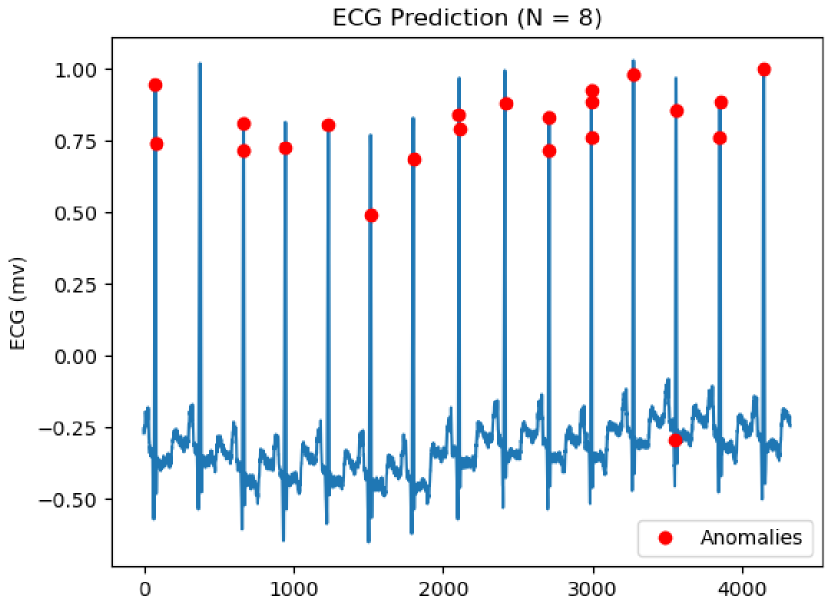

Figure 50: LSTM MLII Prediction W/ Anomaly (N=8)

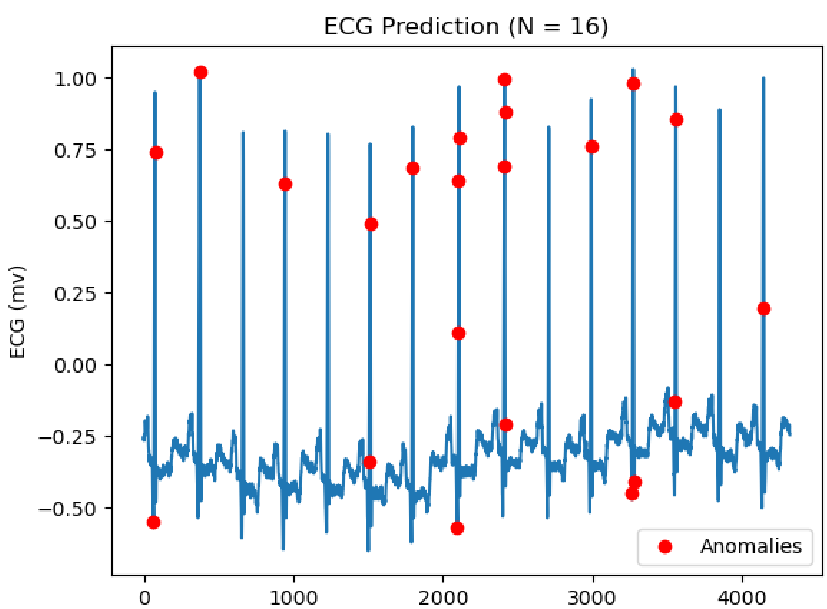

Figure 51: LSTM MLII Prediction W/ Anomaly (N=16)

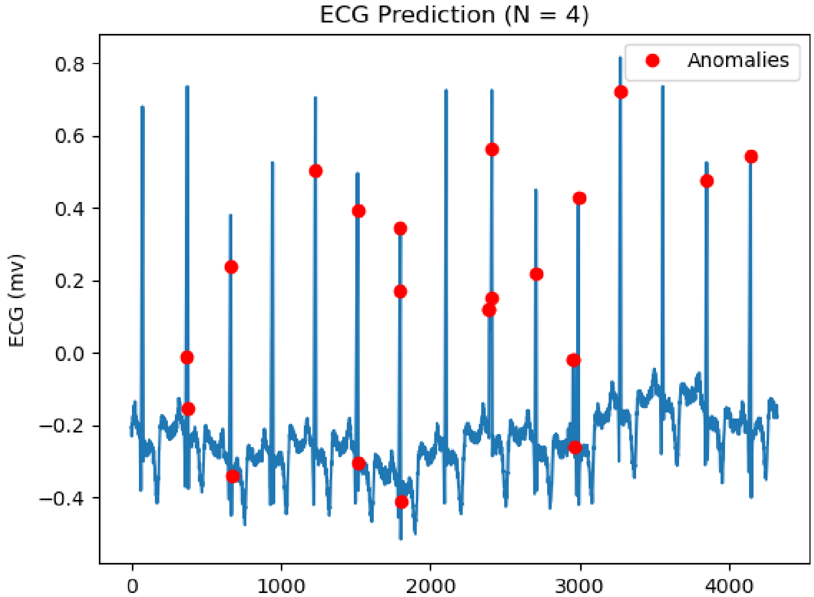

Figure 52: LSTM V5 Prediction W/ Anomaly (N=4)

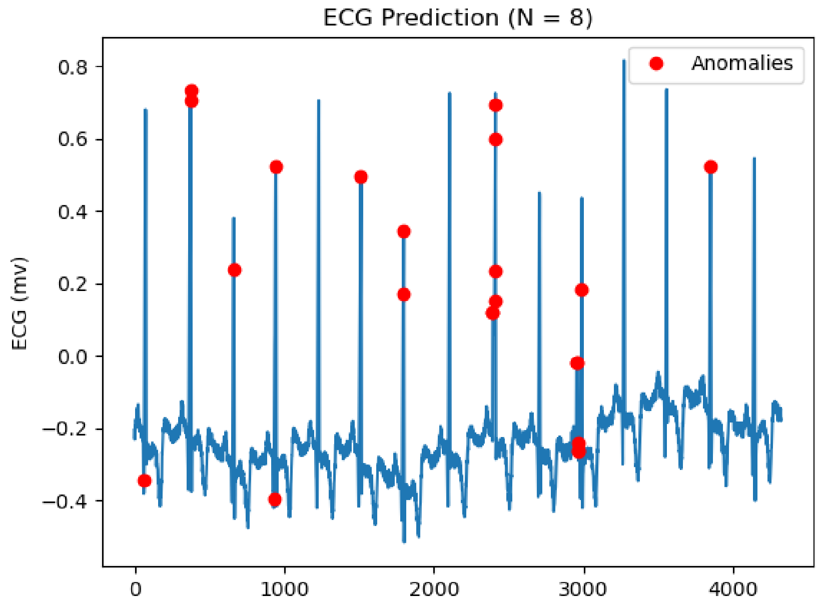

Figure 53: LSTM V5 Prediction W/ Anomaly (N=8)

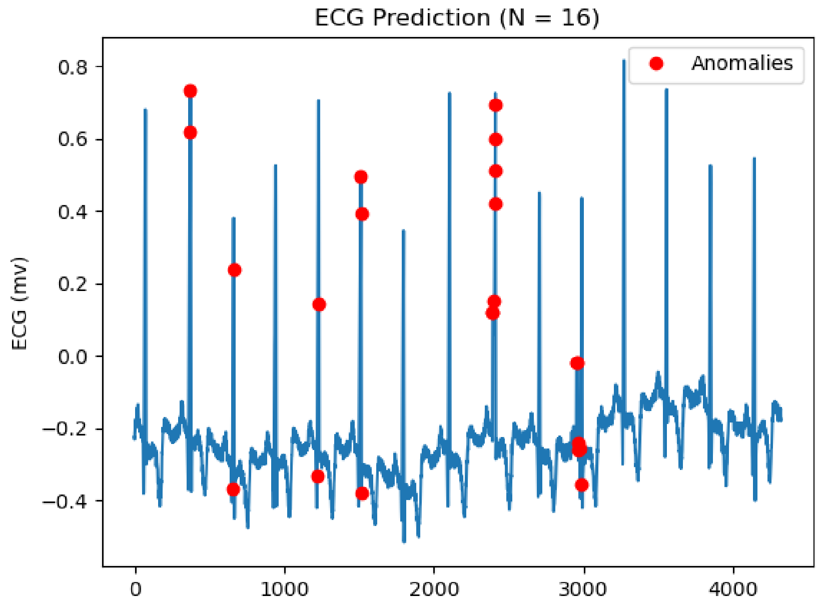

Figure 54: LSTM V5 Prediction W/ Anomaly (N=16)

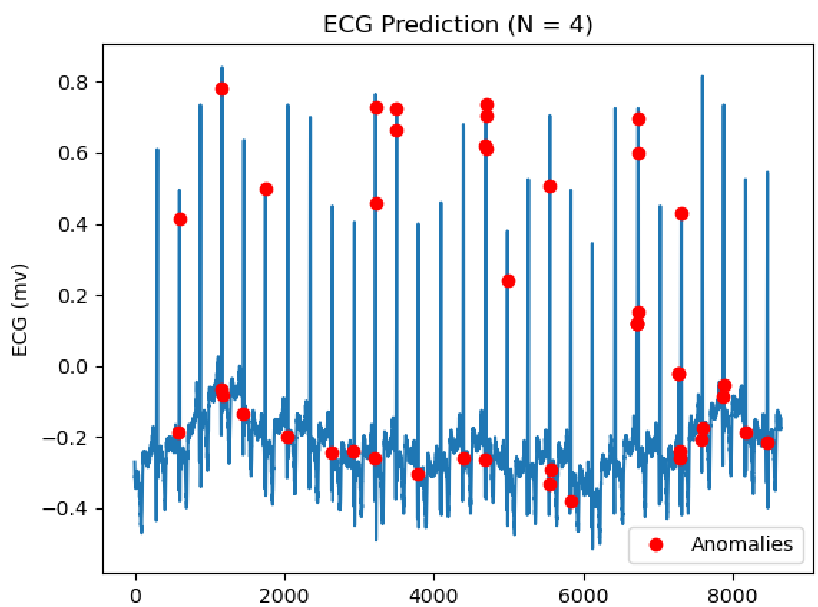

Figure 55: LSTM MLII-V5 Prediction W/ Anomaly (N=4)

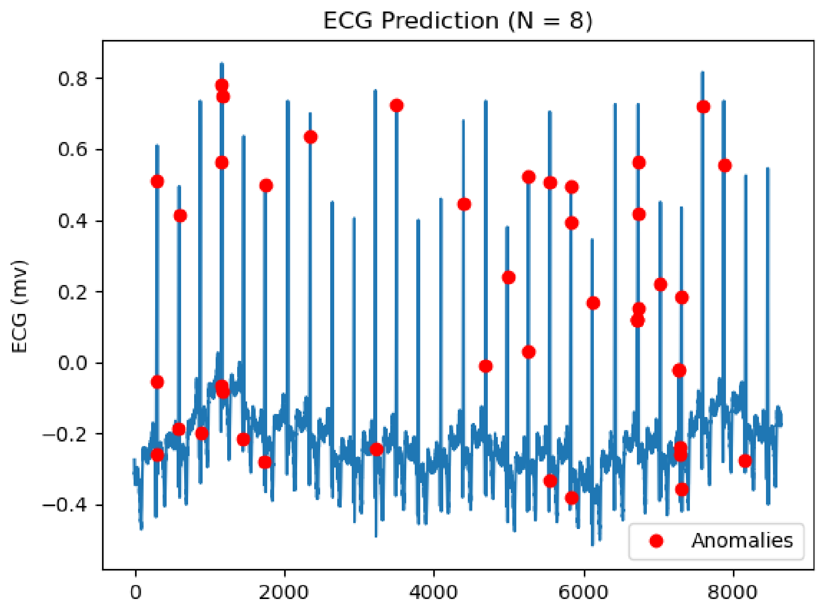

Figure 56: LSTM MLII-V5 Prediction W/ Anomaly (N=8)

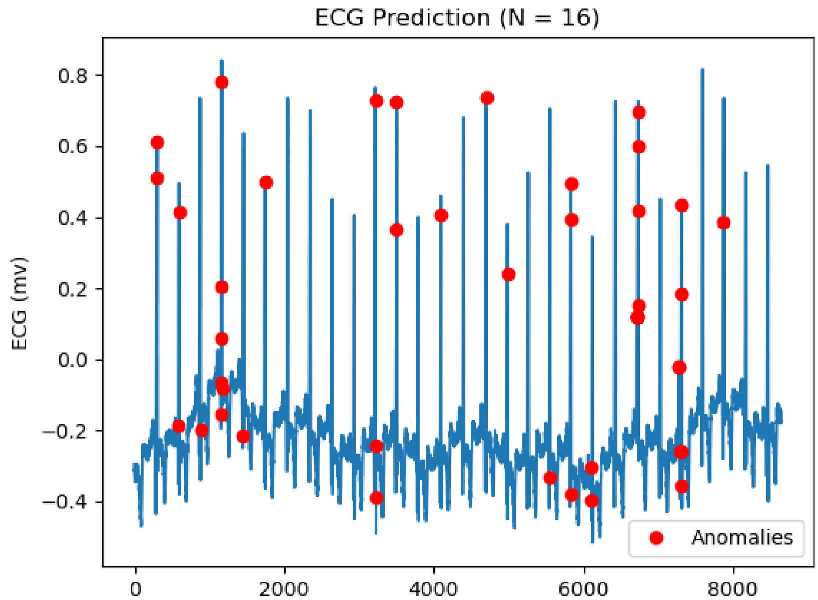

Figure 57: LSTM MLII-V5 Prediction W/ Anomaly (N=16)

Discussion:

1. How have you designed the neural network? What hyper-parameters do you choose? Why?

   The models use an input vector of 2. Loss function is "MSE" and optimizer is "Adam". Activation function is "Relu". Epochs are set at 100. 64 is the number of units of the LSTM.

   Parameters were chosen based on visual inspection of different input vectors. Optimizer, activation function and loss function are kept consistent across the project for better evaluation .

2. Can the neural network fit well to the ECG signals? What is the accuracy of your model?

   The models can fit well into the ECG signals. As noted in the table documenting various accuracy parameters, there is no significant difference between various n value.

3. How much is the influence of the input vector size on the prediction accuracy?

   An increase in n has limited effects on the accuracy of these models. Increase not to be e$^{-}4inscaleanddeemedirrelevant$.

4. How many epochs do you use for training your LSTMs in order to achieve good accuracy? How much is the learning rate? What is your training optimization algorithm (e.g. SGD, Adam etc.)?

   Epochs were set at 100. Optimization algorithm is "Adam" with a default learning rate of 0.01.

5. Which way of treating the time series data gives better accuracy: two uni-variate series or one bi-variate series? Why?

   ECG data of various probes tends to be independent of other leads. Thus, in this case it would make more sense to treat the data as uni-variate so the model can identify anomalies unique to each data set.

4 Clustering-based Anomaly detection

4.1 Anomaly detection by clustering

Generate a two-variable (X1, X2) time series, each variable with 200 data points. X1($\mu = 0$, $\sigma = 2$), X2($\mu = 1$, $\sigma = 2$) are stochastic variables subject to Gaussian distribution. Assume an anomaly ratio of 2%. Employ the K-means clustering algorithm and Self-Organizing Map (SOM) to identify the anomaly points using the clustering-based anomaly detection method.

1. Visualize the multi-variate series, plotting the line plot and scatter plot.

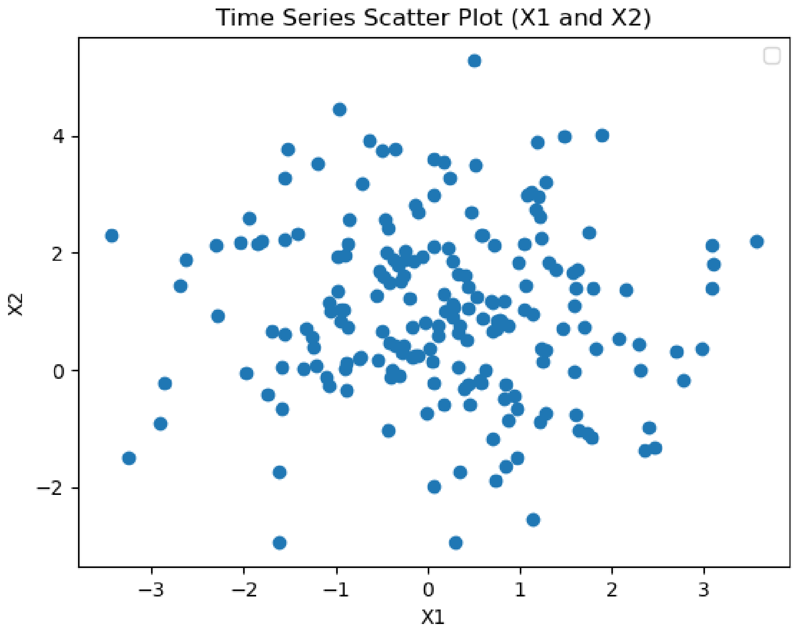

Figure 58: Cluster Time Series Scatter Plot

2. Determine the number of clusters, and do clustering on the data.

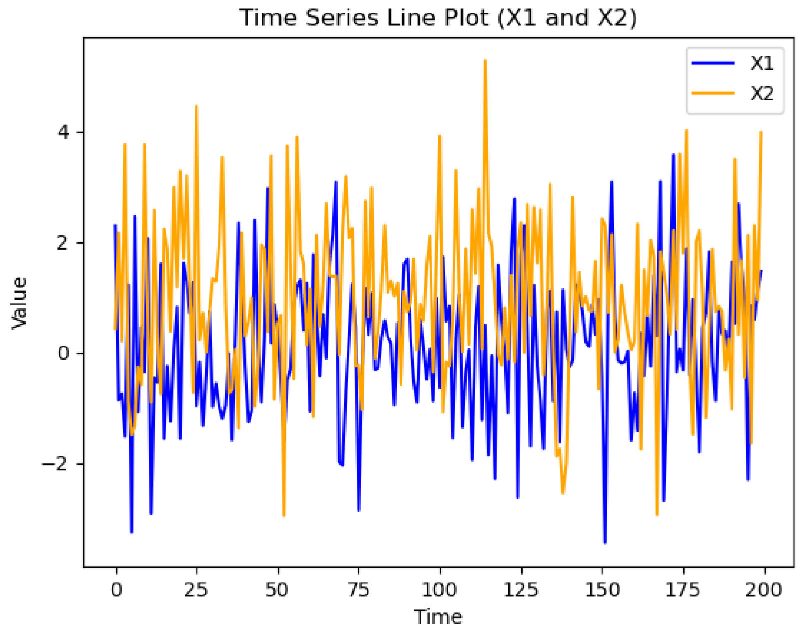

Figure 59: Cluster Time Series Line Plot

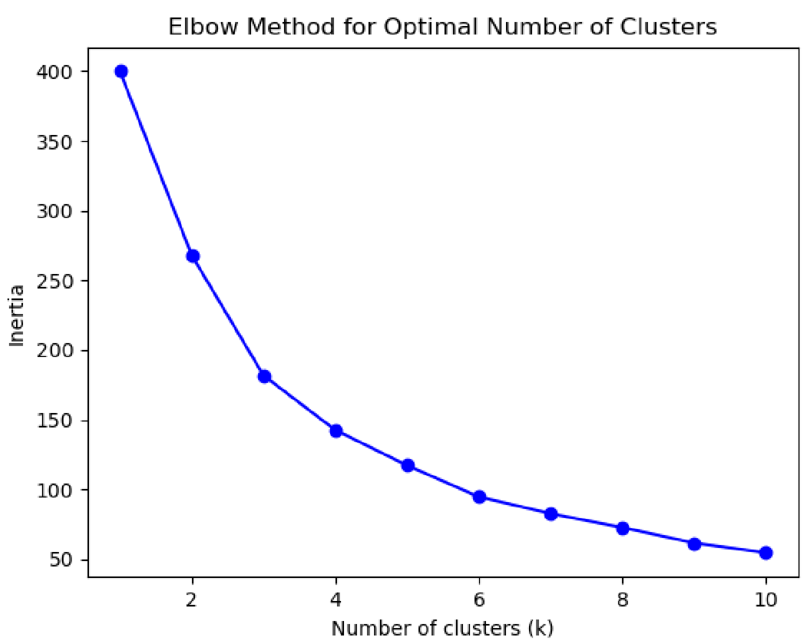

Figure 60: Cluster Elbow

Applied elbow method to identify ideal number of clusters

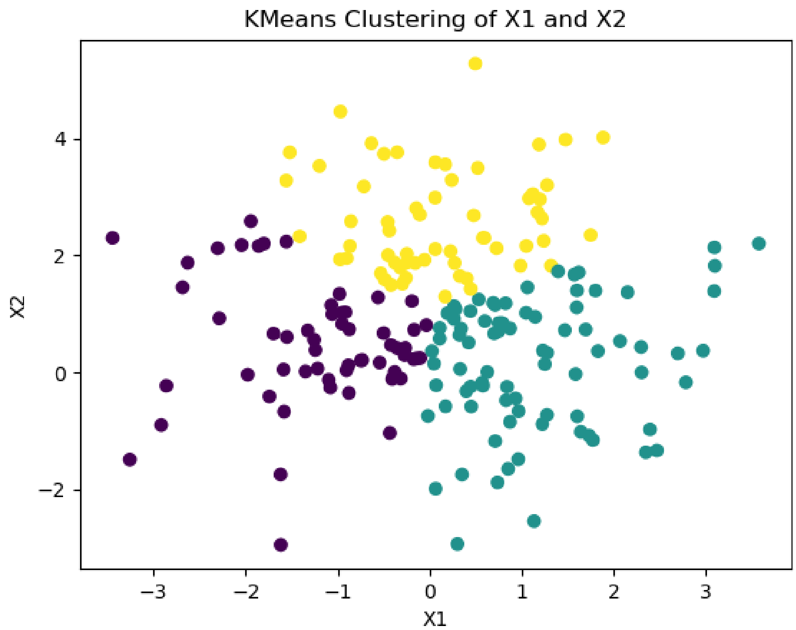

Figure 61: Kmeans Clusters

Figure 62: SOM Clusters

3. Calculate the distance between each point and its nearest centroid (the centroid of its belonging cluster).

4. Use the outlier ratio to calculate the total number of anomalous points (outliers). According to the point-wise distance, those points with largest distances are considered to be the outliers. This step leads naturally to set the distance threshold. By this step, the data points are split into a normal subset and an anomalous subset.

   K-Means Anomaly Positions: [[-2.64627014 -1.6957103] [0.26796137 2.93285798] [-2.79274523 0.8938159] [1.34583122 2.07020401]]

   SOM Anomaly Positions: [[-2.64627014 -1.6957103] [-1.3726856 -2.69299129] [0.26796137 2.93285798] [1.34583122 2.07020401]]

5. Visualize anomalies in a cluster view using the scatter plot (for 2D, 3D data), and in a time series view using the line plot.

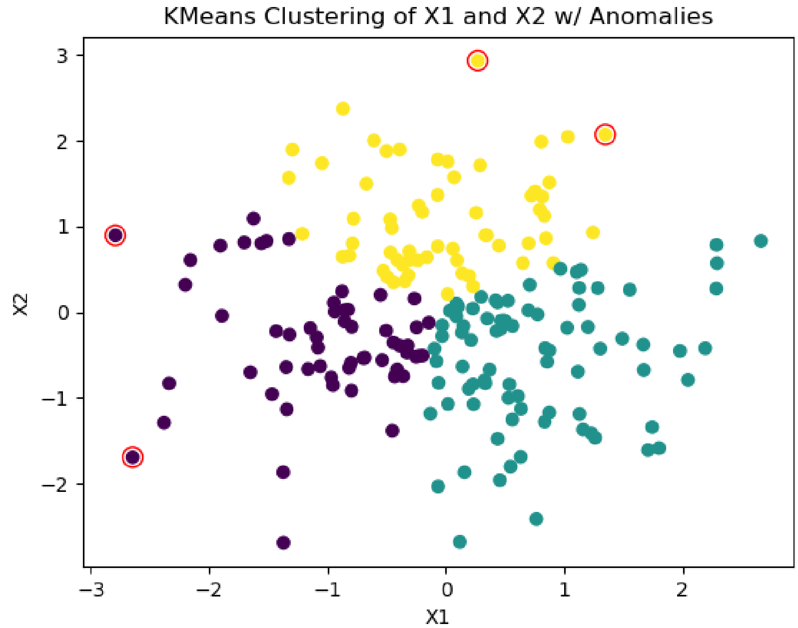

Figure 63: K Means Clusters w/ Anomalies

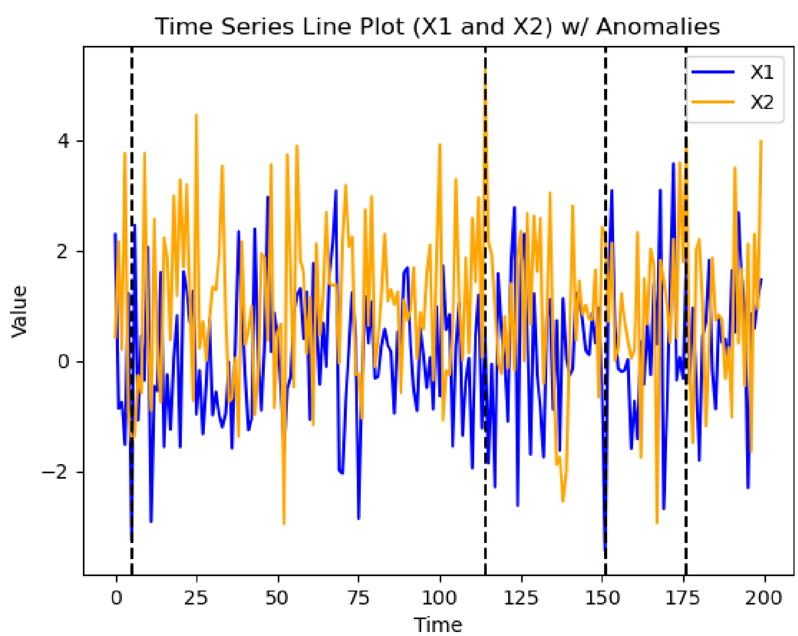

Figure 64: K Means Time Series w/ Anomalies

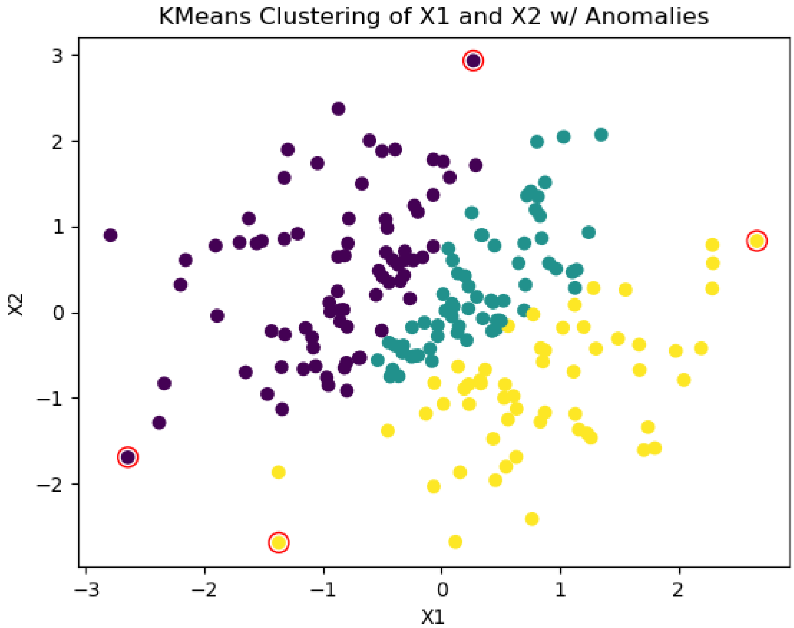

Figure 65: SOM Clusters w/ Anomalies

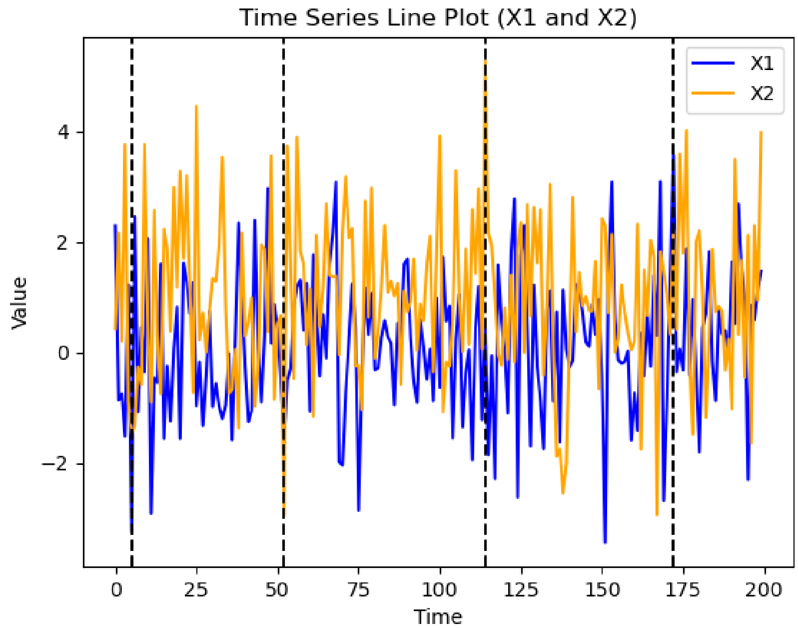

Figure 66: SOM Time Series w/ Anomalies

Discussion:

1. How do you set the number of clusters? Why?

   I used the elbow method to determine the ideal number of clusters. The x value (number of clusters) where the elbow joint is found, refers to the ideal number of clusters.

   The method involves running the clustering algorithm with different values of k and plotting the within-cluster sum of squares (WCSS) against the number of clusters.

2. Which distance metric do you use? Are there other distance metrics which might be useful for this task?

   I used Euclidean distance.

   Alternative methods include: Manhattan distance , Cosine distance, Minkowski distance etc

3. Do the two different clustering methods (K-means and SOM) achieve the same results? Discuss why or why not

   No. 3 out of 4 of the anomalies are identical. However they each choose 1 different anomaly.

This maybe because the anomalies are very close to the threshold. Due to random variations in initialisation of clusters or even mathematical calculations, its possible two similar data points were classified differently.

5 Summary

5.1 Time-series modeling and prediction discussion:

1. Describe the two approaches in your own words.

   Neural Networks (NNs) are machine learning models based on the structure of the human brain. They consist of layers of neurons, where each connection has an associated weight. Each neuron performs some computation on data. NNs can model complex, non-linear relationships in data.

   An ARIMA model is denoted as ARIMA(p, d, q), where. p is the number of lag observations in the AR part. d is the degree of differencing required to make the series stationary. q is the size of the moving average window.

2. Compare the two approaches and discuss their strength and weakness.

   NN's: Pros: Can model non-linear relationships better. Perform better with larger data sets. Require less human intervention.

   NN's Cons: Require Large data sets to train the mode. Computationally intensive. More complex than an ARIMA model. A black box.

   ARIMA Pros: Require smaller data sets to train the model. Less computationally intensive. Based on statiscal methods so gives better insight into how the model is operating.

   ARIMA Cons: Struggle with non-linear data sets. Require human intervention to analyze the data and assign possible p,d,q values. Data must be stationary.

3. List key points of the two approaches and key pros and cons

   Noted above.

   NN: Split data into training and testing subsets. Model training. Model training. Model forecasting. Comparison with test data set.

   Arima: Check stationary. Modify as required. Split data into training and testing subsets. Identify p,d,q values based on acf, pacf and differencing required to achieve stationarity. Model training. Model forecasting. Comparison with test data set.

5.2 Anomaly detection methods discussion:

1. Describe the three different methods.

Decomposition: Data is decomposed into its unique features including trend, seasonality etc. The anomalies are identified from from anlaysing the different features.

Prediction: A model is trained on a subset of the data set. It is then used to predict the data set. A comparison is performed of the actual and predicted dataset. Anomalies are identified based on differences.

Clustering: Involves clustering of all data points based on distance of data to nearest data point. A threshold is identified based on zscore. All data points that fall outside of this threshold are deemed an anomalies.

2. Compare and discuss the strength and weakness of the three methods.

   Decomposition Pro: Visualisation gives insight Feature extraction helps understand underlying patterns Less processing complexity

   Decomposition Cons: Requires human intervention Difficult to analyse with noisy data Assumes seasonality and trends exist

   Prediction Pros: If model is trained correctly, can detect anomalies effectively

   Prediction Cons: Depends a lot on pre-processing of data Requires large data sets

   Clustering Pros: Segments data to for easy interpretation

   Clustering Cons: Heavily dependent on number of clusters High computation for large data sets

3. List key points of the approaches and key pros and cons

   Refer to above.

   Decomposition: Perform exploratory data analysis, visually identification, zscore etc.

   Prediction: Perform exploratory data analysis. Follow steps as mentioned in previous section based on which model is being applied (NN vs ARIMA). Train the model on training subset. Compare model prediction with actual data and identify anomalies.

   Clustering: Identify number of clusters. Cluster data based on distance from centroids. Identify threshold. Identify data points outside of thresholds as anomalies.

4. Find another anomaly detection method from literature and list key pros and cons

   Isolation Forest is a technique for identifying outliers in data that was first introduced by Fei Tony Liu and Zhi-Hua Zhou in 2008. The approach employs binary trees to detect anomalies, resulting in a linear time complexity and low memory usage that is well-suited for processing large datasets. It is a popular machine-learning method for performing unsupervised anomaly detection on multi-dimensional data.

Pros: It does not require any assumptions of data. Efficient. Has a linear time complexity. Has high dimensionality of data

Cons: Heavily dependant on parameters selected Does not require assumptions of data (Unsupervised Learning) and hence can result in false positives.