Neuro-Fuzzy Computing
3rd Problem Set

Ioannis Roumpos 2980
Konstantinos Vermisoglou 2988
Nikos Gkagkosis 3079

March 3, 2024

Contents

1 Problem - 01

From the problem statement, we know that the network will need to have two inputs, and we can use one output to distinguish the two classes. The network should produce a positive output whenever the input vector is in the shaded region(Class I) and a negative output otherwise(Class II). The Class I region is made up of two simple subregions,and it appears that two neurons should be sufficient to perform the classification.

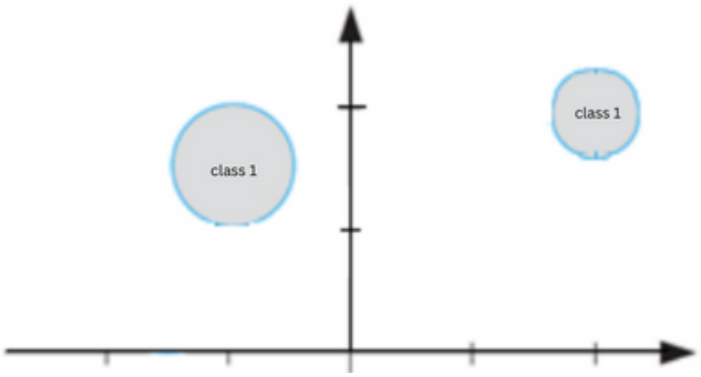


Figure 1

First Layer:

The rows of the first-layer weight matrix will create centers for the two basis functions, and we will choose each center to be located in the middle of one subregion. From the plot we figure that the centers of the two circles are the first (-1 , 1.5) and the second (2 , 2).

$$\mathbf{W}^1 = \begin{bmatrix} -1 & 1.5 \\ 2 & 2 \end{bmatrix}$$

The choice of the biases in the first layer depends on the width that we want for each basis function. The first bias will be smaller than the second bias because, the first basis function should be wider than the second.The boundary formed by the first basis function should have a radius of approximately 1, while the second basis function boundary should have a radius of approximately 0.5. We use a bias of 2 for the second neuron and a bias of 1 for the first neuron.

$$a = e^{-n^2} = e^{-(2 \cdot 0.5)^2} = e^{-1} = 0.3679, a = e^{-n^2} = e^{-(1 \cdot 0.1)^2} = e^{-1} = 0.3679$$

$$\mathbf{b}^1 = \begin{bmatrix} 1 \\ 2 \end{bmatrix}$$

Second layer:

We want the output to be negative for inputs outside the decision regions, so we will use a bias of -1 for the second layer.We choose a value of 2 for the second layer weights, in order to bring the peaks back up to 1.

$\mathbf{W}^2 = \begin{bmatrix} 2 & 2 \end{bmatrix}, \mathbf{b}^2 = [-1]$.

2 Problem - 02

In RBF layer, weights represent a specific location in the input space and we call them centers. This location determines the activation strength of the corresponding basis function in the network.
More centers means more flexibility since they can capture better the input space.
The bias (std dev OR variance OR spread const) performs a scaling operation on the transfer (basis) function, causing it to stretch or compress.

Sampling and Initialization of the RBF network

- Sampling data points with uniform distribution in interval [-4,4] which means that all values within a given range have an equal probability of occurring
- Weights are initialized also random in the interval (-4,4)
- Betas are initialized with value 1
- Learning rate = 0.01

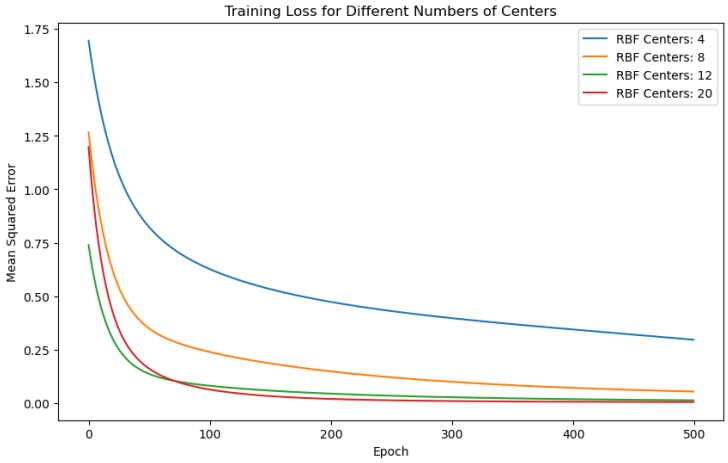


Figure 2: MSE loss for different numbers of centers

MSE is low for every possible choice of number of centers. As centers are increased the loss is decreased.
More significant change is from 4 centers to 8 and from 8 to 12. Changing the number of centers from 12 to 20 does not decrease so much the MSE.

Centers	MSE loss
4	0.296
8	0.053
12	0.012
20	0.004

RBF Network Prediction

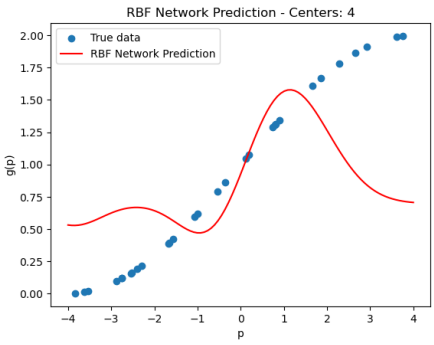


Figure 3: 4 Centers RBF Prediction

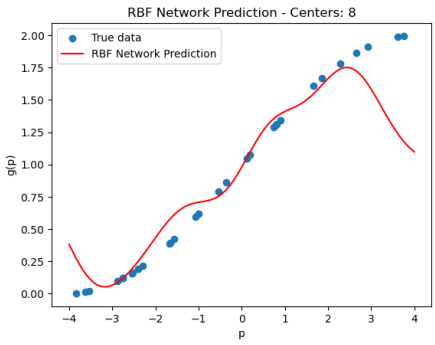


Figure 4: 8 Centers RBF Prediction

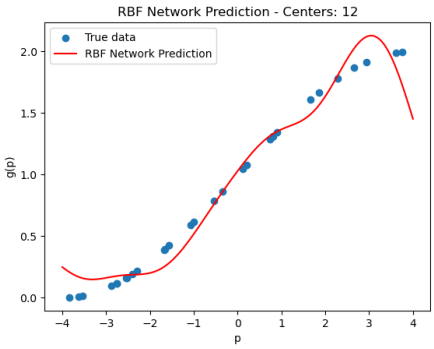


Figure 5: 12 Centers RBF Prediction

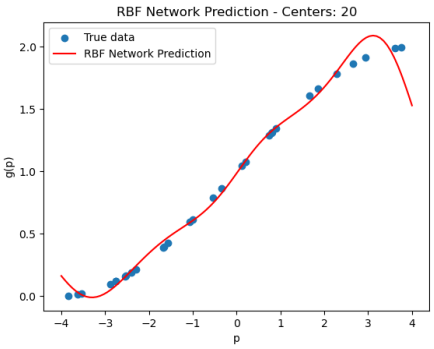


Figure 6: 20 Centers RBF Prediction

4 Centers

Curve does not fit well the data, only in region [-1,1]. 4 centers cannot capture the input space.

8 Centers

Curve captures well the pattern of the function that we try to approximate except the edges of the interval.Also the curve is smooth and not strictly passing through the data points providing a flexibility.

12 Centers

Curve captures even better the pattern of the function and in compare to 8 Centers, in p ¿ 2 the curve is getting very closer to the data points from the curve of 8 Centers.

20 Centers

Curve is even more accurate and in compare to 12 Centers, in p ¡ -2 the curve is getting closer to the data points.

Different Sampling method and Initialization of the RBF network

Lowering the values of biases has as result the centers to be wider and capture more points in the function activation region.
Sampling data points with Gaussian distribution in interval [-4,4] with mean = 0 and std = 1
Weights are initialized also random in the interval (-4,4)
Betas are initialized with value 0.5
Learning rate = 0.01

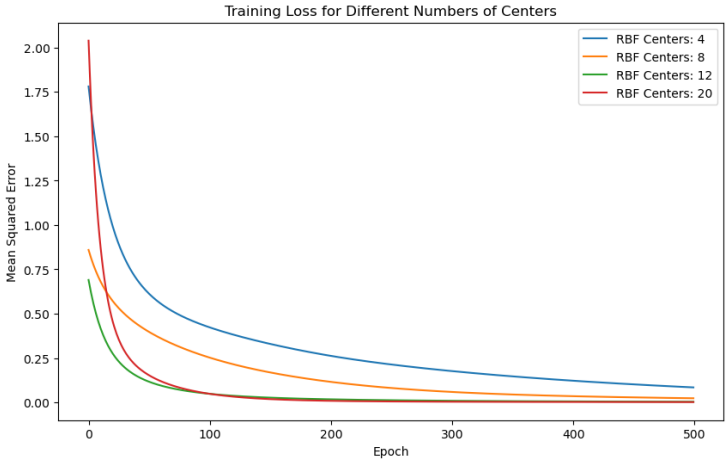


Figure 7: MSE loss for different numbers of centers

MSE is lower than the previous initializations. As centers are increased the loss is decreased. More significant change is from 4 centers to 8 and from 8 to 12. Changing the number of centers from 12 to 20 does not decrease so much the MSE.

Centers	MSE loss
4	0.084
8	0.023
12	0.003
20	0.0017

RBF Network Prediction

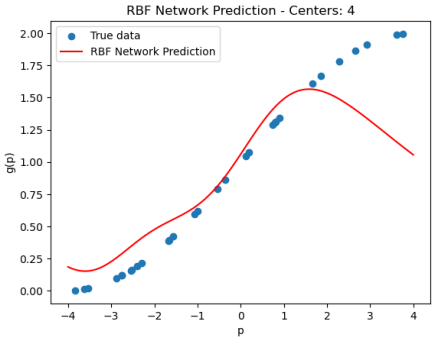


Figure 8: 4 Centers RBF Prediction

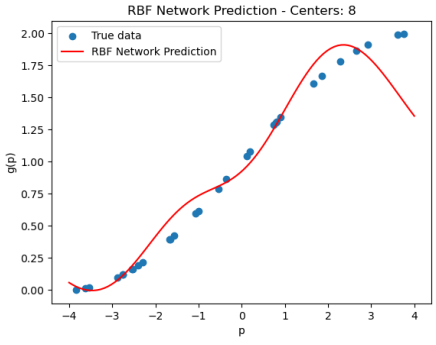


Figure 9: 8 Centers RBF Prediction

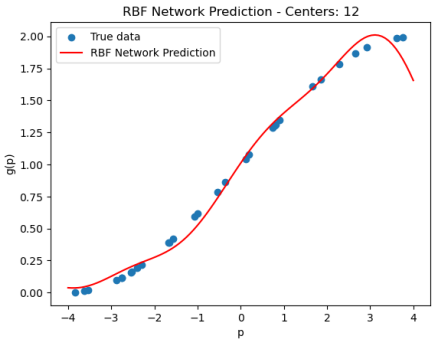


Figure 10: 12 Centers RBF Prediction

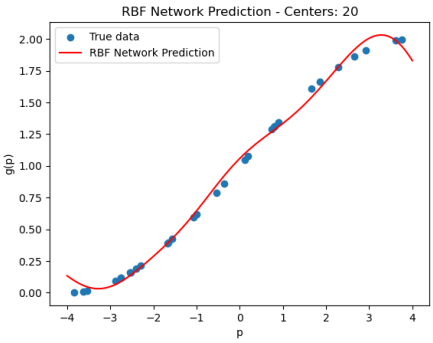


Figure 11: 20 Centers RBF Prediction

4 Centers

Curve fits the data points significantly better than the curve with the previous initialization. It does not fit well after p ¿ 2.

8 Centers

Curve captures better the patterns of the function in regions p ¡ -2 and p ¿ 2 in compare to the previous initialization.

12 Centers

Curve captures even better the pattern of the function and in compare to 8 Centers, in p ¿ 2 the curve is getting very closer to the data points from the curve of 8 Centers.

20 Centers

Curve is even more accurate and in compare to 12 Centers and in compare to the previous initialization. It fits perfect the data points and approximate very efficient the function.

3 Problem - 03

Inside the competitive layer there are two neurons. The neuron's weight vector that is the closest to p, it wins the competition ( ) and has output a 1. After that we apply the learning rule to the winning neuron with a learning rate of 0.5.

Use this technique to train a two-neuron competitive layer on the (non-normalized) vectors below.

$$\mathbf{p}_1 = \begin{bmatrix} 1 \\ 1 \end{bmatrix}, \mathbf{p}_2 = \begin{bmatrix} -1 \\ 2 \end{bmatrix}, \mathbf{p}_3 = \begin{bmatrix} -2 \\ -2 \end{bmatrix}$$

Present the vectors in the following order: $\mathrm{p}_1, \mathrm{p}_2, \mathrm{p}_3, \mathrm{p}_2, \mathrm{p}_3, \mathrm{p}_1$. Here are the initial weights of the network:

$$_1\mathbf{w} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, _2\mathbf{w} = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

After the training the final results are :

$$_1\mathbf{w} = \begin{bmatrix} -1.5625 \\ -0.5625 \end{bmatrix}, {}_2\mathbf{w} = \begin{bmatrix} 1 \\ 0.75 \end{bmatrix}$$

4 Problem - 04

We do the same thing we did in Problem 3 but this time with the following inputs and weights.

$$\mathbf{p}_1 = \begin{bmatrix} 2 \\ 0 \end{bmatrix}, \mathbf{p}_2 = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \mathbf{p}_3 = \begin{bmatrix} 2 \\ 2 \end{bmatrix}$$

We will present the vectors again in the following order: $\mathrm{p}_1, \mathrm{p}_2, \mathrm{p}_3, \mathrm{p}_2, \mathrm{p}_3, \mathrm{p}_1$. Here are the initial weights of the network:

$$_1\mathbf{w} = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, {}_2\mathbf{w} = \begin{bmatrix} -1 \\ 0 \end{bmatrix}$$

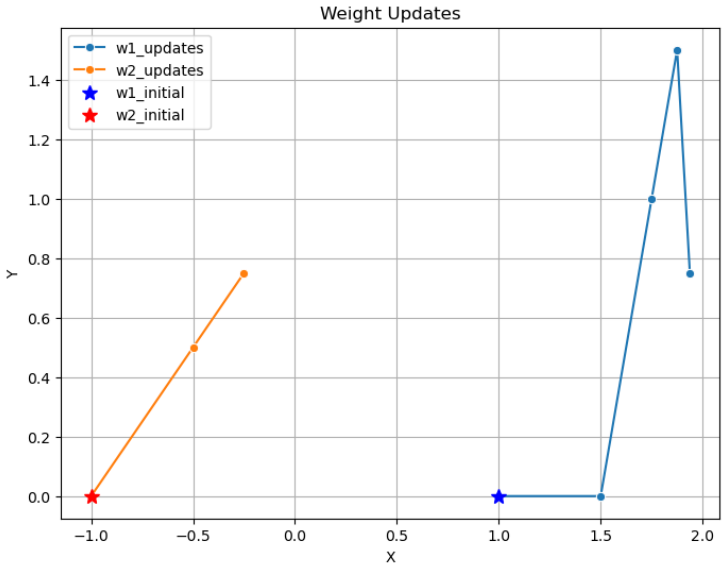


Figure 12: Movements of the weights

During the first few iterations, the winning neuron will depend on the specific presentation order.However, as the number of iterations increases, the network will begin to converge.

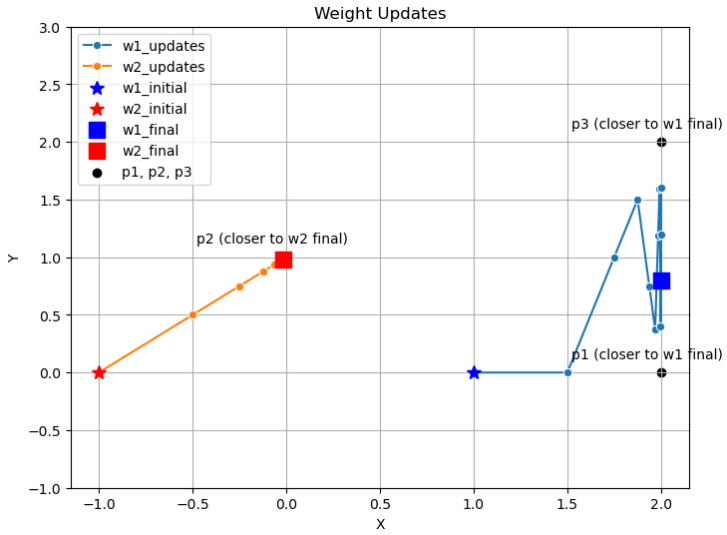


Figure 13: Movements of the weights after a lot iterations

Based on the final values of w1 and w2 after a lot of iterations, it seems like the p1 and p3 will be cluster together and the p2 alone.

5 Problem - 05

A time series of 5000 samples was generated using the Auto regressive model:

$$X_t = a_1 X_{t-1} + a_2 X_{t-2} + a_3 X_{t-3} + U_t$$

with autoregressive coefficients and initial values:

$$a_1 = 0.5$$
$$a_2 = -0.1$$
$$a_3 = 0.2$$
$$U_t = (0, 0.5)$$
$$X_{t-1} = 1$$
$$X_{t-2} = 2$$
$$X_{t-3} = 3$$

A sample of the produced time series in shown below.

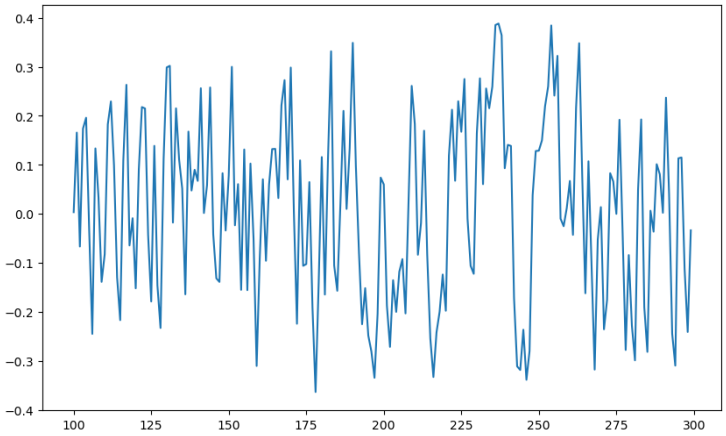


Figure 14: AR time series model

Before building the RNN model we must identify the order of the AR model. To do this we plot autocorrelation and partial autocorrelation functions to analyze the correlation structure.
The coefficient of correlation between two values in a time series is called the autocorrelation function (ACF) For example the ACF for a time series y_t is given by:

$$corr(y_t, y_{t-k}), k = 1, 2, ...$$

This value of k is the time gap being considered and is called the lag. A lag 1 autocorrelation for example is the correlation between values that are one time period apart.

Autocorrelation

The ACF is a way to measure the linear relationship between an observation at time t and the observations at previous times.

Partial Autocorrelation

If we assume an AR(k) model, then we may wish to only measure the association between y_t and y_{t-k} and filter out the linear influence of the random variables that lie in between them. By calculating the correlation of the transformed time series we obtain the partial autocorrelation function (PACF).

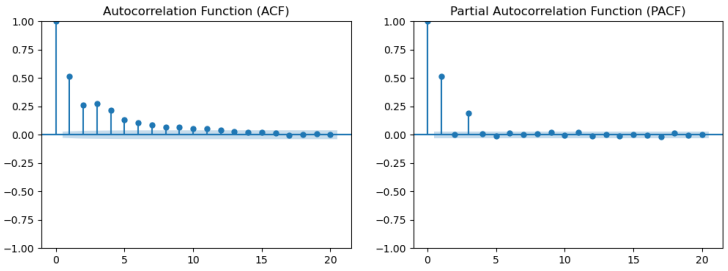


Figure 15: ACF and PACF of AR model

The PACF is most useful for identifying the order of an autoregressive model. A sudden cutoff suggests that the direct relationship exists only up to that lag. So based on the PACF plot, AR(1), AR(2) seem a good order.

However this visualization does not mean that necessarily determines the order of the AR model. To determine the order of the AR model we evaluate in the RNN model the mse of test set for order = 1, 2, 3 and keep the order with the least error.

Model

From 5000 samples we keep 70% for training, 10% for validation, 20% for testing.

Layer1

LSTM layer with 50 neurons and relu as activation function

Layer2

LSTM layer with 50 neurons and relu as activation function

Layer3

Full connected layer with one output neuron corresponding to the X_t

Order	RNN MSE
1	0.022
2	0.0219
3	0.0212

So the time steps = 3 in the RNN model.

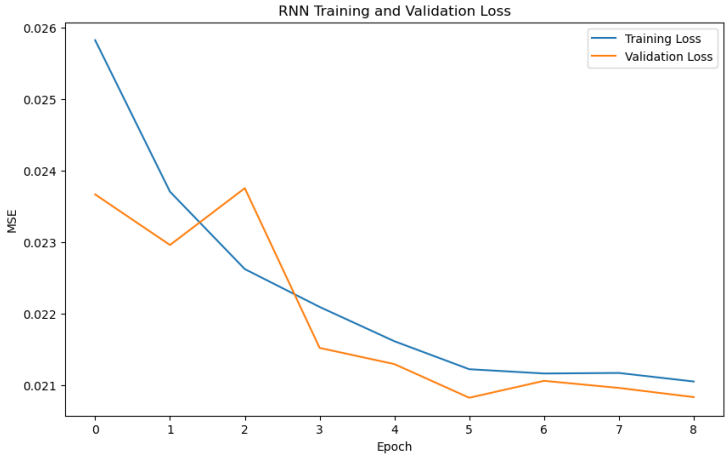


Figure 16: MSE loss of RNN

The validation loss is not constantly decreased during epochs but generally MSE loss is pretty good.
The RNN prediction of the time series in a sample of the original time is shown below:

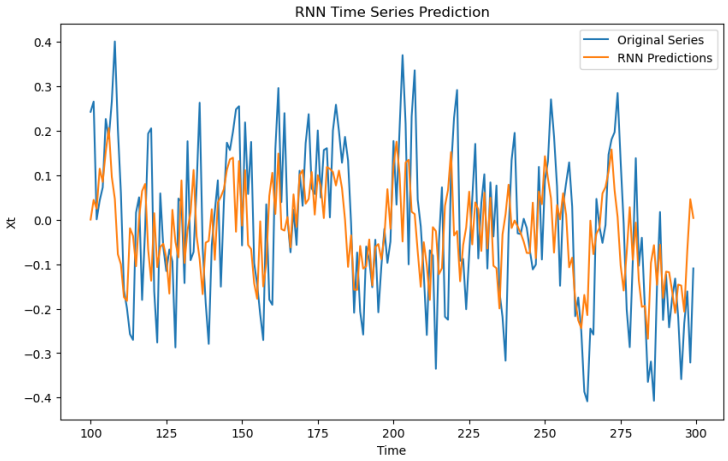


Figure 17: RNN Prediction of time series

The prediction of the AR model using the RNN model is very accurate and can capture the pattern of the series.

6 Problem - 06

Moving Average

The moving average models the relationship between the current value and past white noise error terms. An MA(q) model expresses the current value as a linear combination of past q error terms. The order q indicates how many past error terms are considered.

A time series of 5000 samples was generated using a moving average model:

$$X_t = U_t + a_1 U_{t-1} + a_2 U_{t-2} + a_3 U_{t-3} + a_4 U_{t-4} + a_5 U_{t-5} + a_6 U_{t-6}$$

$$a_1 = a_2 = 5$$
$$a_3 = a_4 = a_5 = a_6 = -0.5$$
$$U_t = (0, 0.5)$$

A sample of the produced time series is shown below:

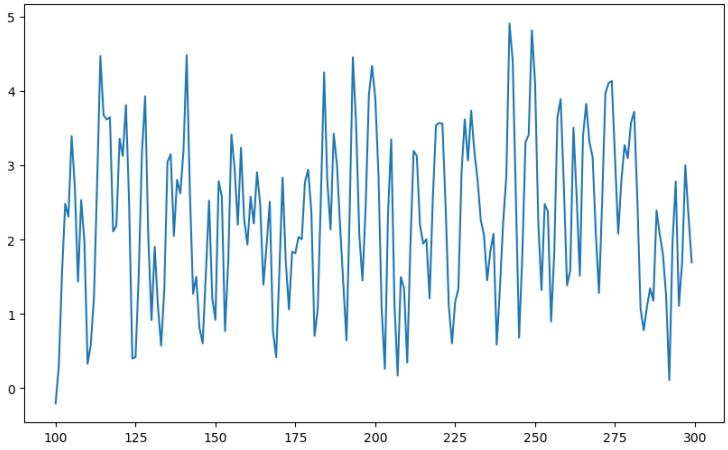


Figure 18: AM time series model

We follow exactly the same process as problem 5:

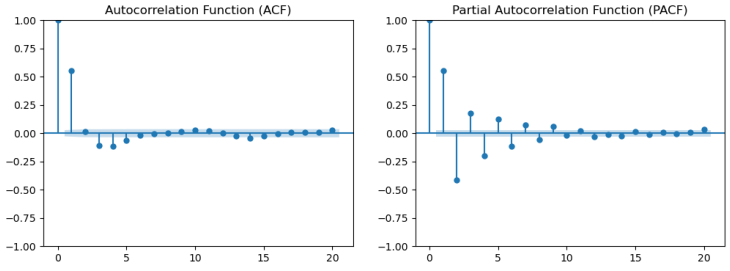


Figure 19: ACF and PACF of time series

From the PACF orders from 1 to 4 seem a good value for the time steps of the RNN model. To determine the order of the AM model we evaluate in the RNN model the mse of test set for order = 2, 3, 4, 5, 6 and keep the order with the least error.

Order	RNN MSE
2	0.685
3	0.624
4	0.582
5	0.577
6	0.534

So the time steps = 6 in the RNN model.

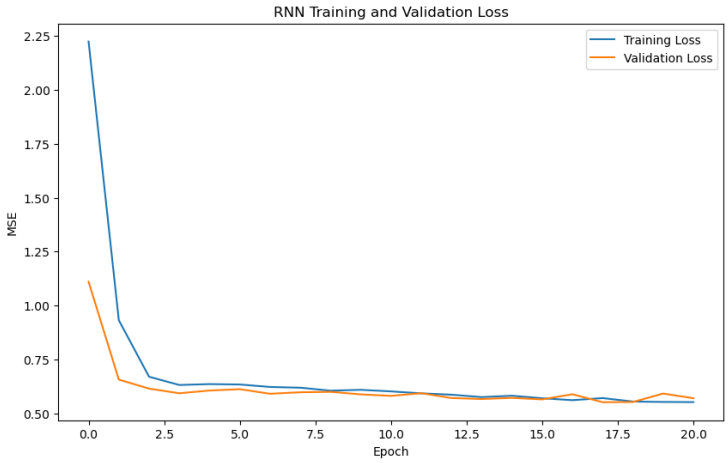


Figure 20: RNN MSE loss

The validation loss is not constantly decreasing but in average the MSE loss is low.
The RNN prediction of the time series in a sample of the original time is shown below:

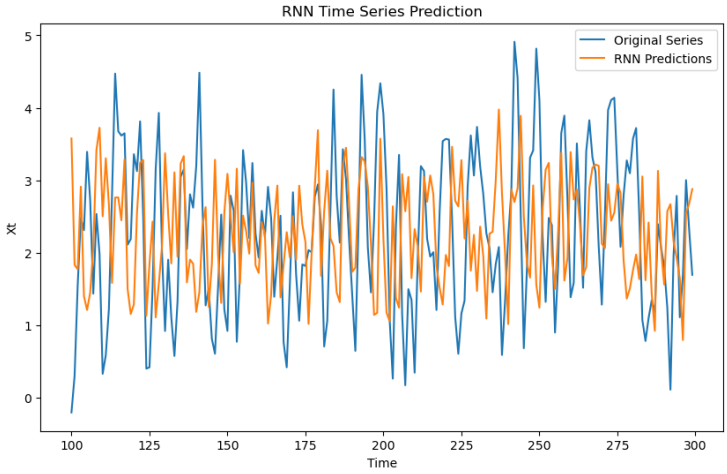


Figure 21: RNN MSE loss

The prediction of the AM model using the RNN model is very accurate and can capture the pattern of the series.
MSE of AR model is slightly lower than MSE of AM model. This can be due to the fact that AR model consider the relationship between the current observation and its past values while AM capture the relationship between the current value of the series and the past q errors. AR models are best suited for modeling time series data that exhibit autocorrelation. On the other hand, MA models are best suited for modeling time series data that exhibit moving average patterns, where the value of the series at a given time is influenced by past errors.

7 Problem - 07

For the given expression we will define the next membership functions.

a) $\mu_a(x) = max(0, 1 - \frac{|x-15|}{15})$

b) $\mu_b(x) = e^{-x^2}$

c) $\mu_c(x) = max(min(\frac{x-35}{15}, 1, \frac{120-x}{20}, 0))$

d) $\mu_d(x) = \begin{cases} 0 & x \leq 10 \\ \frac{x-10}{5} & 10 \leq x \leq 15 \\ \frac{20-x}{5} & 15 \leq x \leq 20 \\ 0 & x \geq 20 \end{cases}$

8 Problem - 08

From the theory in the lectures we have that:

$$\mu_{\tilde{R}_1}(x, y) \geq 0.3 \tag{1}$$

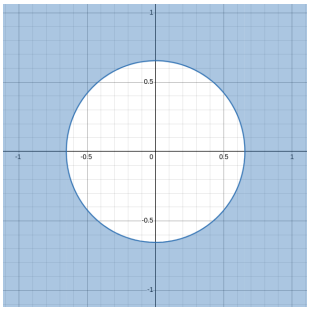
$$1 - \frac{1}{1 + x^2 + y^2} \geq 0.3 \tag{2}$$

$$\frac{1}{1 + x^2 + y^2} \leq 0.7, (1 + x^2 + y^2 > 0) \tag{3}$$

$$x^2 + y^2 \geq \frac{3}{7} \tag{4}$$

The equation $x^2 + y^2 = \frac{3}{7}$ represents a circle centered at the origin with radius $\sqrt{\frac{3}{7}}$.
We draw the circle on the xy-plane.

$$x^2 + y^2 = \frac{3}{7}$$



The inequality $x^2+y^2 \geq \frac{3}{7}$ represents all the points inside or on the boundary of this circle.

We can express the solution set as the region inside or on the circle, or in set notation as

$$\{(x, y) \mid x^2 + y^2 \geq \frac{3}{7}\}$$

9 Problem - 09

We define a ordinary set $N(\tilde{A})$ which is nearest to the fuzzy subset $\tilde{A}$ and has the following membership function: $\mu_{N(\tilde{A})}(x) = \begin{cases} 0 & \mu_{\tilde{A}}(x) \leq 0.5 \\ 1 & \mu_{\tilde{A}}(x) \geq 0.5 \end{cases}$
Given that we conclude on the following expressions that will help us find the index of fuzziness.

$$\left|\mu_{\tilde{A}}(x) - \mu_{N(\tilde{A})}(x)\right| = \mu_{\tilde{A}-\tilde{A}^C}(x) = \left|\mu_{\tilde{A}}(x) - \mu_{\tilde{A}_{0.5}}(x)\right|$$

$$v(\tilde{A}) = \frac{2}{n}\sum_{i=1}^{n}\left|\mu_{\tilde{A}}(x_i) - \mu_{N(\tilde{A})}(x_i)\right| = \frac{2}{n}\sum_{i=1}^{n}\left|\mu_{\tilde{A}}(x_i) - \mu_{\tilde{A}_{0.5}}(x_i)\right|$$

To approximate the above sum we can integrate for the interval of the given membership function as described by the following expression for the first part of the problem **A**.

$$v(\tilde{A}) = \frac{\int_0^a (\frac{x^2}{a^2})^2\,dx}{(\int_0^a \frac{x^2}{a^2}\,dx)^2} = \frac{3}{5}$$

For the second part **B** the index is calculated as follows:

$$v(\tilde{A}) = \frac{\int_0^{a/2} (\frac{x^2}{a^2})^2\,dx}{(\int_0^{a/2} \frac{x^2}{a^2}\,dx)^2} + \frac{\int_{a/2}^a (\frac{(x-a)^2}{a^2})^2\,dx}{(\int_{a/2}^a \frac{(x-a)^2}{a^2}\,dx)^2} = \frac{6}{5}$$

10 Problem - 10

The max-min decomposition between 2 fuzzy subsets $\tilde{R_1}$ and $\tilde{R_2}$ is defined as follows:

$$\mu_{\tilde{R_1}\circ\tilde{R_2}} = \bigvee[\mu_{\tilde{R_1}}(x,y) \bigwedge \mu_{\tilde{R_2}}(y,z)] = max[min\{\mu_{\tilde{R_1}}(x,y), \mu_{\tilde{R_2}}(y,z)\}]$$

In the next step we want to find the min of the membership functions assuming the y variable is fixed.

$$f = \mu_{\tilde{R_1}}(x,y) = e^{-k(x-y)^2}$$
$$g = \mu_{\tilde{R_2}}(y,z) = e^{-k(y-z)^2}$$

Calculating the derivative we can find the min x_0 for each function and from there the point where that happens is the min value $f(x_0) = min$.

$$\frac{df}{dx} = -2ke^{-k(x-y)^2}(x-y) = 0 \tag{5}$$
$$-2ke^{-k(x-y)^2} < 0 \rightarrow x - y = 0 \rightarrow x = y \tag{6}$$
$$\frac{dg}{dx} = -2ke^{-k(y-z)^2}(y-z) = 0 \tag{7}$$
$$-2ke^{-k(y-z)^2} < 0 \rightarrow y - z = 0 \rightarrow y = z \tag{8}$$

From the equations above we get that the min point is when $x = y$ and $y = z$ and we can get the min value.

$$f = \mu_{\tilde{R_1}}(x,x) = e^{-k(y-y)^2} = e^0 = 1$$
$$g = \mu_{\tilde{R_2}}(z,z) = e^{-k(z-z)^2} = e^0 = 1$$

So we conclude that $min(f,g) = min(1,1) = 1$ and $max = 1$ and $\mu_{\tilde{R_1}\circ\tilde{R_2}} = 1$

The figure below shows the graphically representation and we can see the max value at 1.

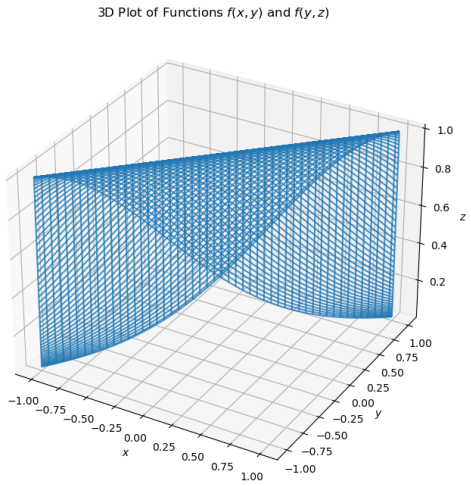


Figure 22: Composition of 2 subsets