

รายงานการทดลอง

Computer Assignment 3

นายธนวัฒน์ บำเพ็ญพันธุ์ 630610736

CPE 261456

(Introduction to Computational Intelligence)

ภาคเรียนที่ 1 ปีการศึกษา 2565

การทดลองการจำแนก Class

บน Wisconsin Diagnostic Breast Cancer (WDBC) Dataset

การทดลองการจำแนก Class มะเร็งเต้านม โดยรับ Input เป็นตัวเลขข้อมูล 30 inputs มาจำแนก Class โดยมี Output 2 แบบ คือผลการวินิจฉัยโรค M = Malignant (มีเนื้อร้าย), กับ B = Benign (ไม่มีเนื้อร้าย) ซึ่งจะใช้ Multilayer Perceptron ที่จะ Train ด้วย Genetic Algorithm ในการจำแนก Class

1.1 การเตรียม Multilayer Perceptron และ Dataset

1.1.1 สร้าง Multilayer Perceptron ที่มีรูปแบบดังนี้

Model		
No.	Layers and Nodes	Activation Function
1.	30-4-1	Unit step
2.	30-8-1	Unit step
3.	30-8-4-1	Unit step
4.	30-8-8-1	Unit step

ตารางที่ 1: รายละเอียดของ Multilayer Perceptron แต่ละตัวที่ใช้ในการทดลอง

Multilayer Perceptron แต่ละตัวใช้ค่า Bias เท่ากับ 0.5 ในแต่ละ Neurons และ Weights ที่สุ่มในตอนเริ่มต้นจะอยู่ในช่วง $[-1,1]$

สมการของ Activation function (Unit step)

$$f(x) = 0 \text{ when } x \leq 0, \\ 1 \text{ when } x > 0$$

1.1.2 จัดการ Dataset

แปลงค่า Output ใน Dataset โดยการแทนด้วยตัวเลขคือ M (Malignant) แทนด้วยเลข 1 และ B (Benign) แทนด้วยเลข 0 และทำการ Normalize ข้อมูลใน Dataset ด้วยวิธี Min-Max Normalization และทำการแบ่งชุดข้อมูล Training และ Validation ใช้สำหรับการทำ 10% Cross validation

สมการ Min-Max Normalization เมื่อ x คือข้อมูลเดิมในแต่ละค่าใน wdbc.data

$$x' = \frac{x - \min(x)}{\max(x) - \min(x)}$$

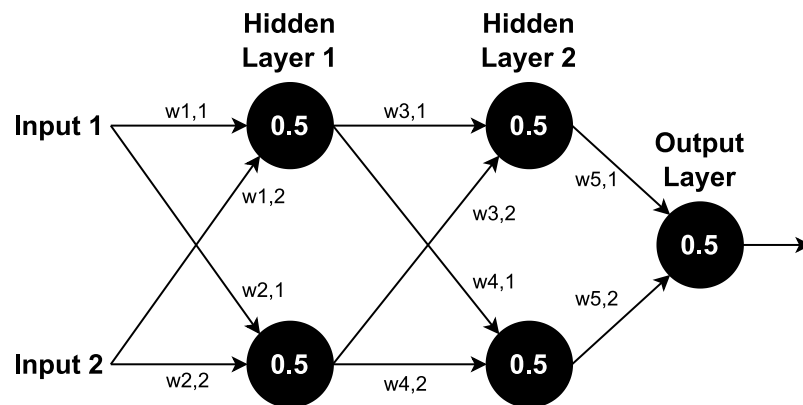
1.2 การ Train ด้วย Genetic Algorithm

Parameter	Value
Population size	20
Maximum generation	100
Crossover	5-points
Mating probability	0.5
Mutation Probability	0.1
Elitism	1

ตารางที่ 2: แสดงค่าของ Parameter แต่ละตัวที่ใช้ในกระบวนการ Genetic Algorithm

1.2.1 การสร้างประชากรเริ่มต้น (Initialize the population)

ทำการนำค่า Weights ของแต่ละ Neurons ใน Multilayer Perceptron ตั้งแต่ Hidden Layer แรกจนถึง Output Layer มารวมกันให้เป็นสายโครโมโซมจะได้เป็น Individual 1 ตัว ตัวอย่างการสร้างตามรูปที่ 1. โดยสร้างทั้งหมด 20 ตัวตาม Population size ที่กำหนดไว้ในตารางที่ 2.



สายโครโมโซม [w1,1 , w1,2 , w2,1 , w2,2 , w3,1 , w3,2 , w4,1 , w4,2 , w5,1 , w5,2]

รูปที่ 1: แสดงตัวอย่าง Individual ที่ได้จากการรวม Weights ใน Multilayer Perceptron เป็นสายโครโมโซม

1.2.2 การคำนวณค่าความเหมาะสม (Fitness value) ของ Individual แต่ละตัว

ค่า Fitness ที่ได้จาก Individual แต่ละตัวจะมาจากการที่ให้ Multilayer Perceptron ที่มี Weights ตามสายโครโมโซม Individual ประมวลผลบน Training Dataset ที่เราจัดการไว้แล้วนำค่า Mean Square Error (MSE) ที่ได้ออกมาคำนวณโดยมีสมการดังนี้เมื่อ I คือ Individual ตัวนั้นๆ

$$Fitness(i) = 1 - MSE_i$$

จะเห็นได้ว่าหาก Individual ตัวนั้นมีค่า MSE ที่น้อยจะทำให้มีค่า Fitness สูง

ทำการ Normalization fitness ให้อยู่ในเรนจ์ [0,1] เพื่อนำไปใช้ในการดำเนินการคัดเลือก

1.2.3 การดำเนินการคัดเลือก (Selection)

จะใช้วิธีการคัดเลือกตามสัดส่วนแบบล้อรูเล็ต (Roulette Wheel Standard) โดยจะหมุนล้อรูเล็ต 20 ครั้งเพื่อสำเนาโครโมโซมไปที่ประชากรกลาง

1.2.4 การดำเนินการการข้ามสายพันธุ์ (Crossover)

สร้าง Mating Pool จากประชากรกลางโดยการสุ่มจับคู่พ่อ-แม่ โดยมีความน่าจะเป็นของการจับคู่ (Mating probability) เท่ากับ 0.5 ตามตารางที่ 2 ซึ่งหากจำนวนโครโมโซมที่ถูกเลือกมาจากประชากรกลางเป็นเลขคี่จะทำการสุ่มเพิ่มมาจากประชากรกลางอีก 1 ตัว ทำการข้ามสายพันธุ์โดยการ

นำพ่อแม่ใน Mating Pool มาข้ามสายพันธุ์ 5 ตำแหน่งตามที่กำหนดไว้ในตารางที่ 2 จะได้ลูกซึ่งจะเป็นสมาชิกที่เป็นไปได้ของประชากรในรุ่นถัดไป และโครโมโซมพ่อแม่จะถูกนำออกจากประชากรกลาง

1.2.5 การดำเนินการการกลายพันธุ์ (Mutation)

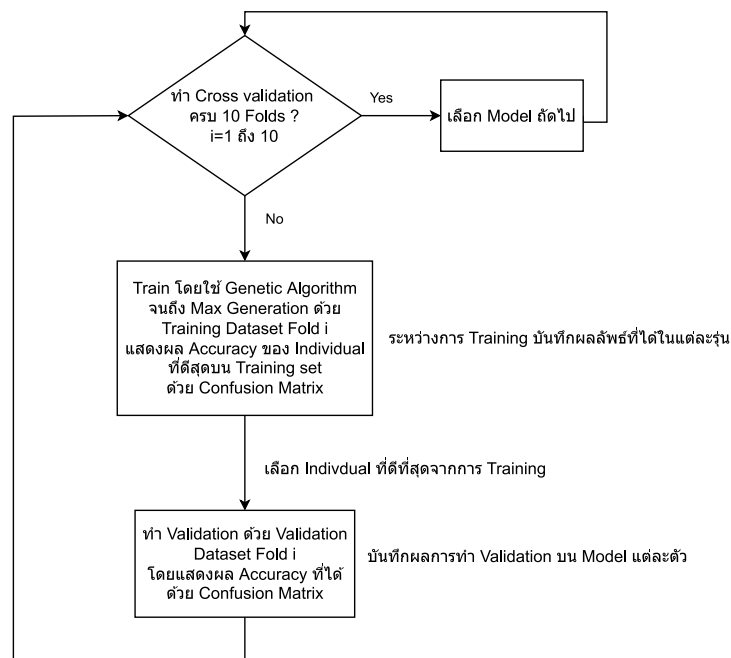
ใช้วิธีการกลายพันธุ์แบบแข็ง (Strong mutation) บนสมาชิกที่เป็นไปได้ของประชากรในรุ่นถัดไป โดยมีการกำหนดค่าความน่าจะเป็นของการกลายพันธุ์ (Mutation probability) เท่ากับ 0.1 ตามที่กำหนดไว้ในตารางที่ 2

1.2.6 การดำเนินการกระบวนการอภิชนนิยม (Elitism)

จะทำการเลือก Individual ที่มีค่า fitness สูงสุด 1 ตัวแล้วใส่เข้าไปในสมาชิกที่จะเป็นประชากรในรุ่นถัดไป

1.3 กระบวนการทดลอง

การทดลองโดยใช้ 10% Cross validation กับ Multilayer Perceptron แต่ละตัวในตารางที่ 1 โดยมีภาพรวมของกระบวนการดังนี้

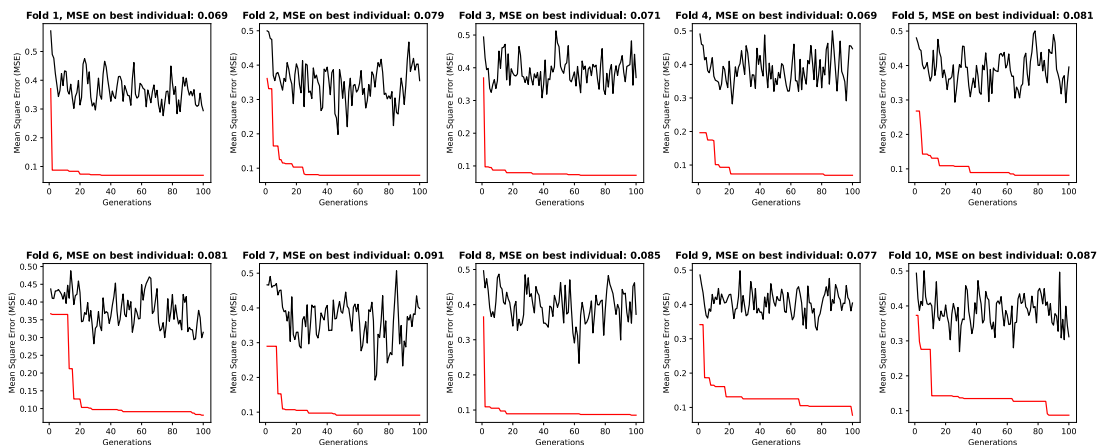


รูปที่ 2: แสดงภาพรวมกระบวนการทดลอง

1.4 ผลการทดลอง

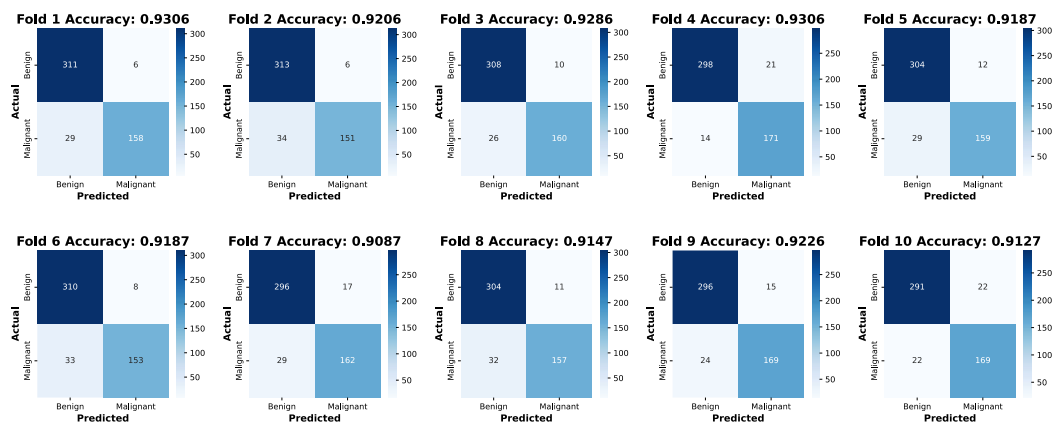
1.4.1 Model ที่ 1. [30-4-1]

**MLP 30-4-1 train with GA
MSE Converge**



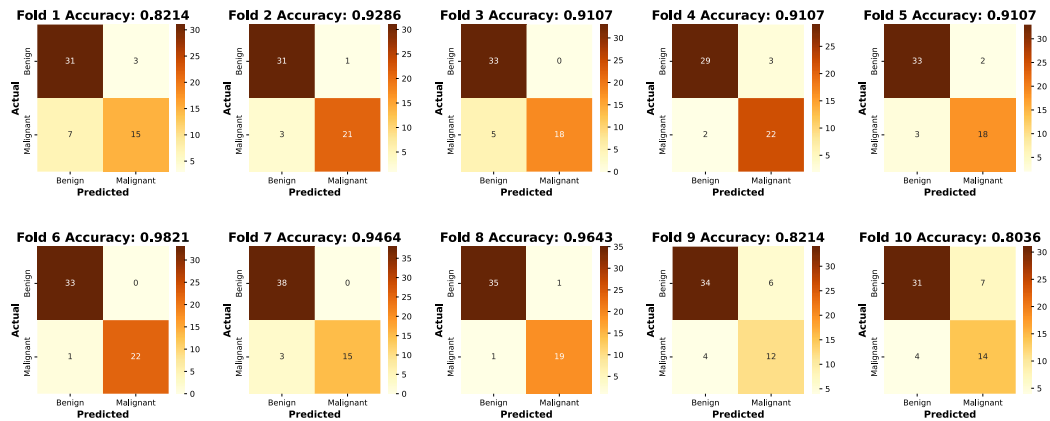
รูปที่ 3: แสดงการลู่เข้าของ MSE บน Model 30-4-1 โดยที่เส้นสีดำคือค่าเฉลี่ยของ MSE จากทุก Individual ณ รุ่นนั้นๆ และ เส้นสีแดงคือ MSE ของ Individual ที่มีค่า Fitness สูงสุด ณ รุ่นนั้นๆ

**MLP 30-4-1 train with GA
Confusion Matrix (Training)**



รูปที่ 4: แสดง Accuracy ที่ได้บน Training set ในแต่ละ Fold ของ Model 30-4-1

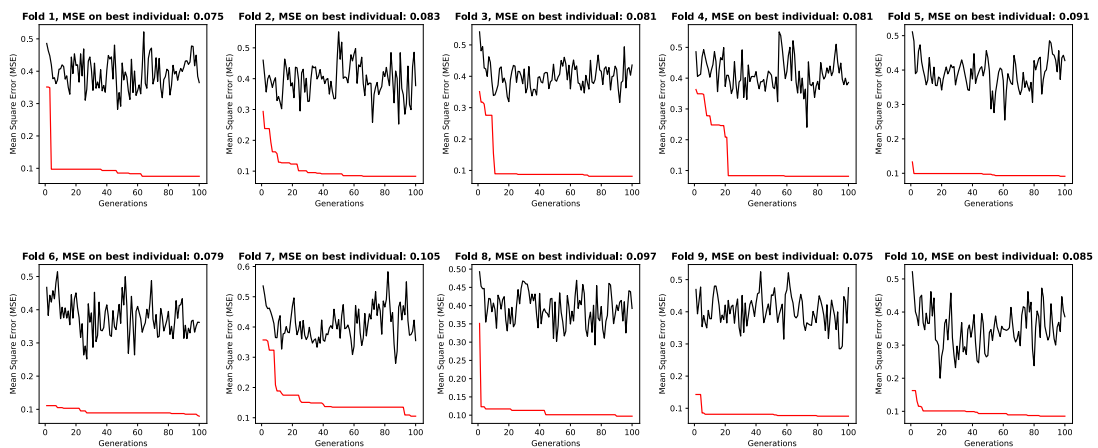
MLP 30-4-1 train with GA Confusion Matrix (Validation)



รูปที่ 5: แสดง Accuracy ที่ได้บน Validation set ในแต่ละ Fold ของ Model 30-4-1

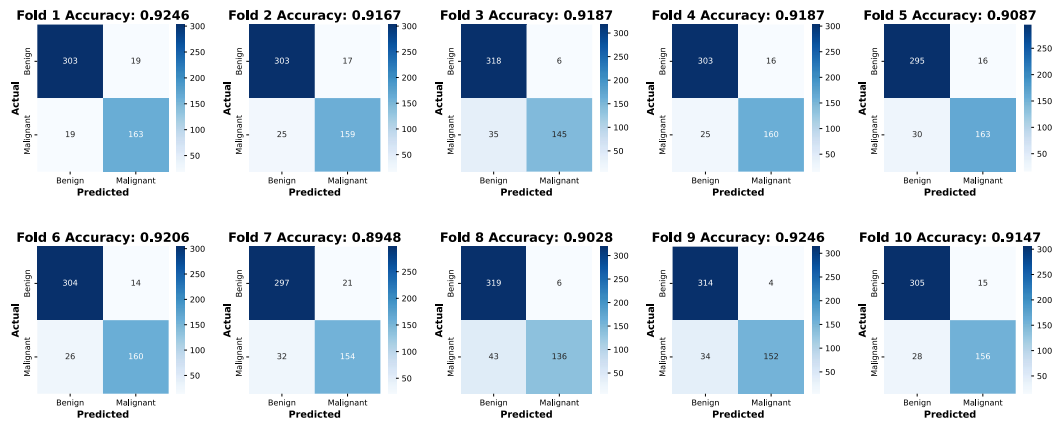
1.4.2 Model ที่ 2. [30-8-1]

MLP 30-8-1 train with GA MSE Converge



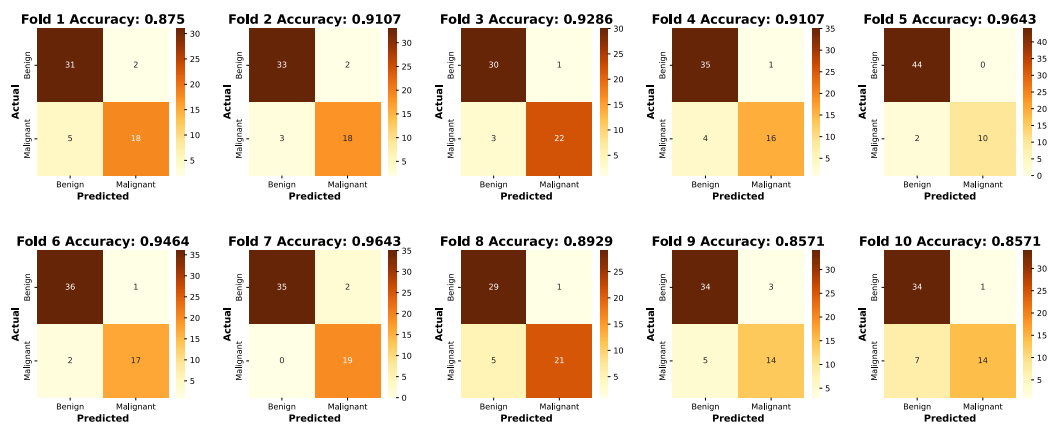
รูปที่ 6: แสดงการลู่เข้าของ MSE บน Model 30-8-1 โดยที่เส้นสีดำคือค่าเฉลี่ยของ MSE จากทุก Individual ณ รุ่นนั้นๆ และ เส้นสีแดงคือ MSE ของ Individual ที่มีค่า Fitness สูงสุด ณ รุ่นนั้นๆ

MLP 30-8-1 train with GA Confusion Matrix (Training)



รูปที่ 7: แสดง Accuracy ที่ได้บน Training set ในแต่ละ Fold ของ Model 30-8-1

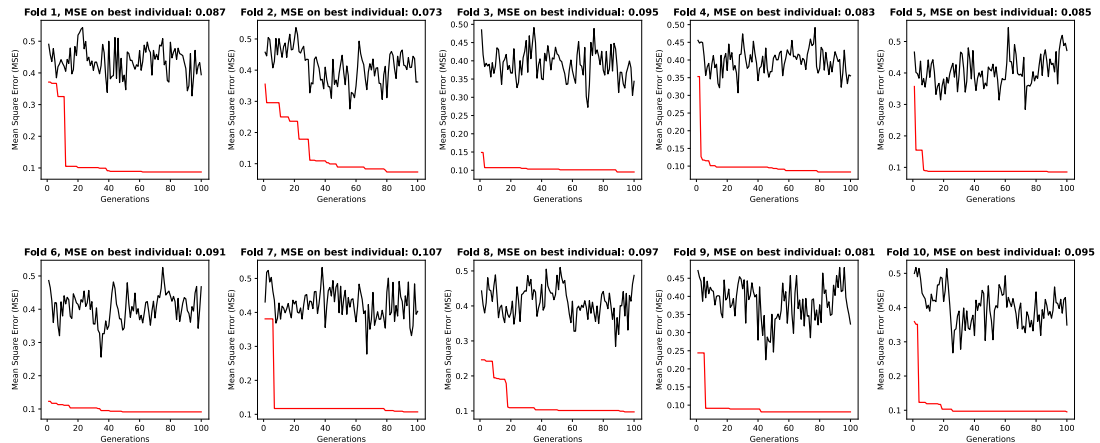
MLP 30-8-1 train with GA Confusion Matrix (Validation)



รูปที่ 8: แสดง Accuracy ที่ได้บน Validation set ในแต่ละ Fold ของ Model 30-8-1

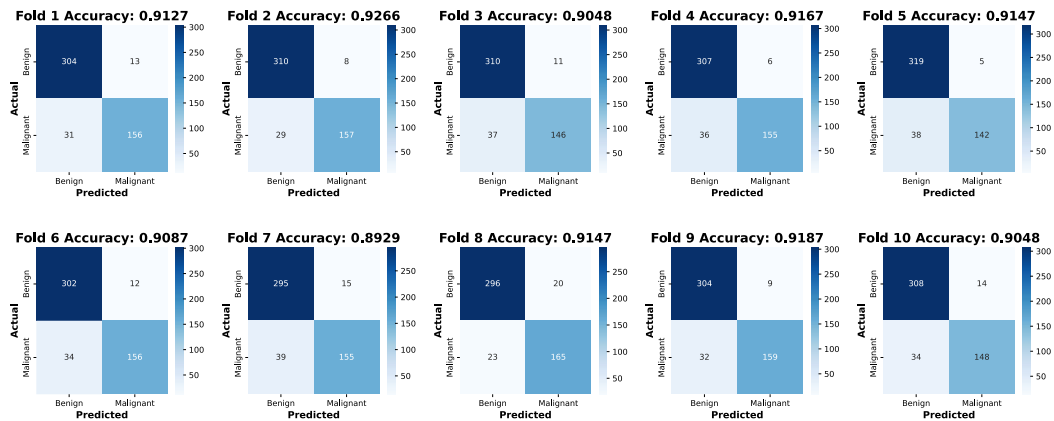
1.4.3 Model ที่ 3. [30-8-4-1]

MLP 30-8-4-1 train with GA MSE Converge



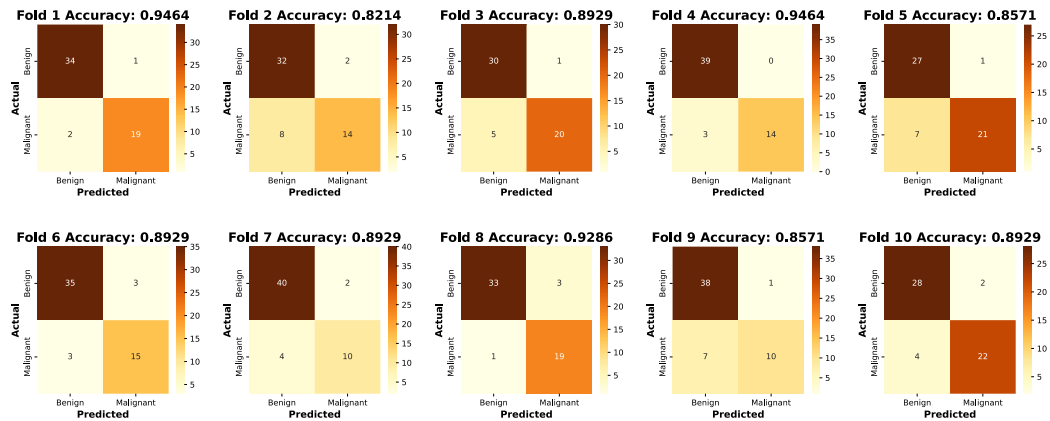
รูปที่ 9: แสดงการลู่เข้าของ MSE บน Model 30-8-4-1 โดยที่เส้นสีดำคือค่าเฉลี่ยของ MSE จากทุก Individual ณ รุ่นนั้นๆ และ เส้นสีแดงคือ MSE ของ Individual ที่มีค่า Fitness สูงสุด ณ รุ่นนั้นๆ

MLP 30-8-4-1 train with GA Confusion Matrix (Training)



รูปที่ 10: แสดง Accuracy ที่ได้บน Training set ในแต่ละ Fold ของ Model 30-8-4-1

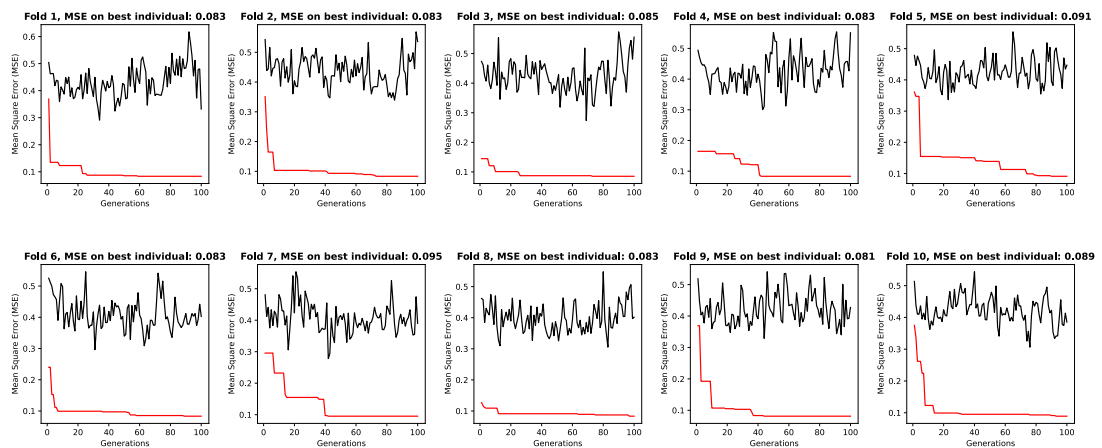
MLP 30-8-4-1 train with GA Confusion Matrix (Validation)



รูปที่ 11: แสดง Accuracy ที่ได้บน Validation set ในแต่ละ Fold ของ Model 30-8-4-1

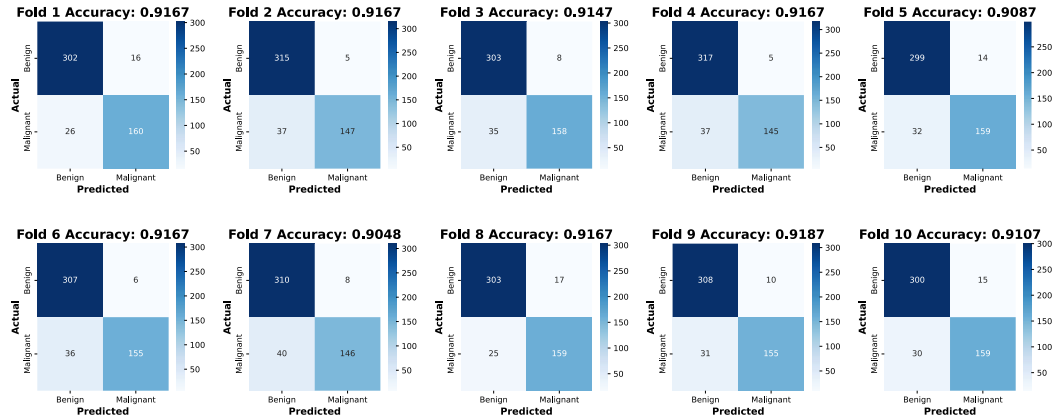
1.4.4 Model ที่ 4. [30-8-8-1]

MLP 30-8-8-1 train with GA MSE Converge



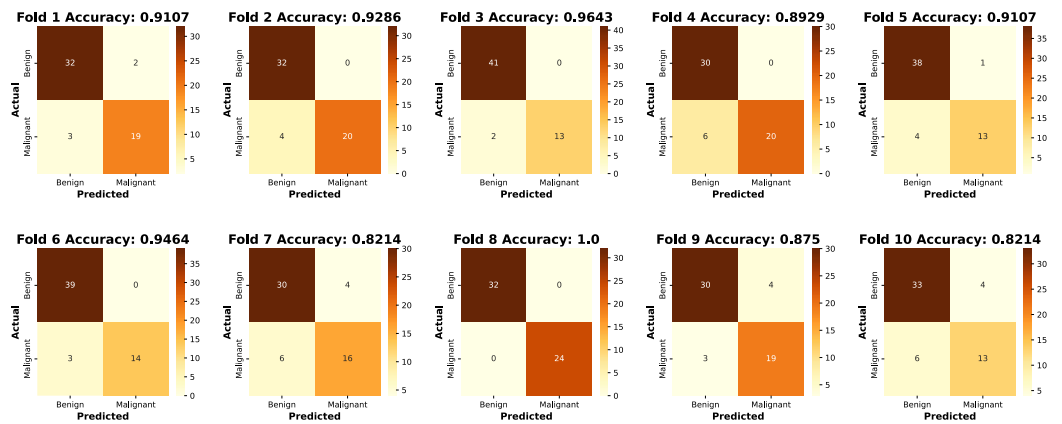
รูปที่ 12: แสดงการรู้เข้าของ MSE บน Model 30-8-8-1 โดยที่เส้นสีดำคือค่าเฉลี่ยของ MSE จากทุก Individual ณ รุ่นนั้นๆ และ เส้นสีแดงคือ MSE ของ Individual ที่มีค่า Fitness สูงสุด ณ รุ่นนั้นๆ

MLP 30-8-8-1 train with GA Confusion Matrix (Training)



รูปที่ 13: แสดง Accuracy ที่ได้บน Training set ในแต่ละ Fold ของ Model 30-8-8-1

MLP 30-8-8-1 train with GA Confusion Matrix (Validation)



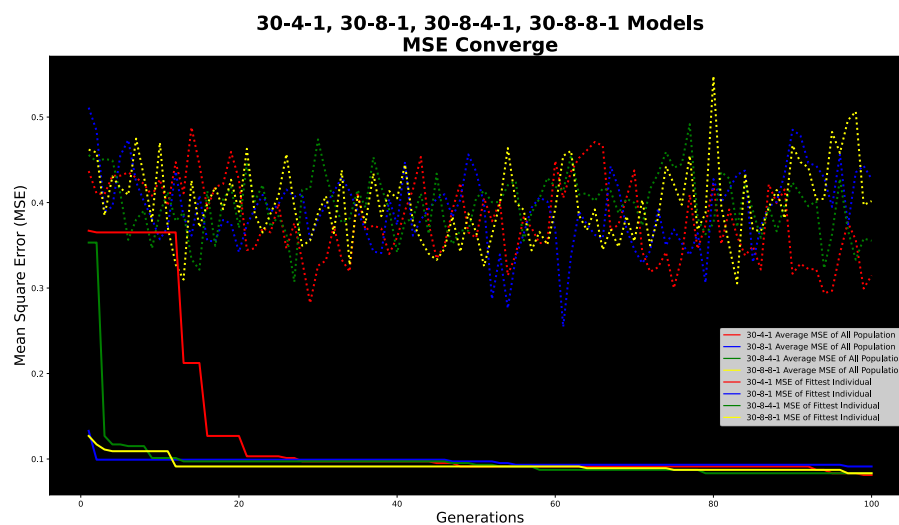
รูปที่ 14: แสดง Accuracy ที่ได้บน Validation set ในแต่ละ Fold ของ Model 30-8-8-1

1.5 สรุปผลการทดลอง

Accuracy เฉลี่ยที่ได้จากการ Training และ Validation แต่ละ Model ในตารางที่ 1.

Model		Accuracy	
No.	Layers and Nodes	Training	Validation
1.	30-4-1	0.92065	0.89999
2.	30-8-1	0.91449	0.91071
3.	30-8-4-1	0.91153	0.89286
4.	30-8-8-1	0.91411	0.90714

ตารางที่ 3: แสดง Accuracy เฉลี่ยบน Training/Validation set ในแต่ละ Model



รูปที่ 15: เส้นประแสดงค่าเฉลี่ยของ MSE ในทุก Individual ของทุก Model ณ รุ่นนั้นๆ
เส้นทึบแสดงค่า MSE ของ Individual ที่ค่า Fitness สูงสุดของทุก Model ณ รุ่นนั้นๆ

1.6 วิเคราะห์ผลการทดลอง

จากตารางที่ 3. จะเห็นว่า Model ที่ 2. 30-8-1 มี Accuracy เฉลี่ยที่ Validation set สูงที่สุดและใกล้เคียงกับ Training set แต่ Accuracy ที่ได้ก็ห่างจาก Model ตัวอื่นๆไม่มากนัก และ Individual ที่ดีที่สุดหลังจาก Train จนถึงรุ่นสุดท้ายแล้วจากทุก Model จะมีค่า Accuracy อยู่ที่ประมาณ 0.91-0.92 จากรูปที่ 15. จะเห็นว่า MSE ของ Individual ที่ดีที่สุดในแต่ละรุ่น มีค่าพอๆกันแล้ว Converge อย่างรวดเร็ว แล้วในช่วงรุ่นที่ 40 เป็นต้นไปเราเห็นการเปลี่ยนแปลงของ Individual ที่ดีที่สุดน้อยลงอย่างมาก และเมื่อดูในช่วงรุ่นที่ 1 ถึง 20 จะเห็นว่ากราฟ Converge ไวมาก ซึ่งอาจเป็นเพราะข้อเสียของการคัดเลือกตามสัดส่วนแบบ Roulette wheel และการส่งต่อ Elitism คือเกิดการลู่เข้าก่อนเวลาอันควรนั่นเอง จึงทำให้ไม่เจอการเปลี่ยนแปลงที่อาจทำให้ได้เจอ Individual ที่ทำให้ Accuracy สูงขึ้นได้ และการเลือกใช้ Unit step function เป็น Activation function ให้กับ Multilayer Perceptron อาจไม่ได้ให้ผลดีและเห็นการเปลี่ยนแปลงเมื่อทำการปรับเปลี่ยนจำนวน Hidden Layer และ Hidden neurons มากนัก