



دانشگاه صنعتی امیرکبیر  
(پای تکنیک تهران)

## پاسخ تمرین پنجم شبکه های عصبی

استاد درس جناب آقای دکتر صفابخش

تدریس یار جناب آقای مهندس اسدی

نیمسال دوم سال تحصیلی ۱۴۰۰-۱۴۰۱

محسن عبادپور

شماره دانشجویی: ۴۰۰۱۳۱۰۸۰

ایمیل: [m.ebadpour@aut.ac.ir](mailto:m.ebadpour@aut.ac.ir)

### فهرست پاسخ ها

۲	توضیحات اولیه، بارگزاری داده ها و پیش پردازش
۳	مسئله اول: قسمت اول
۸	مسئله اول: قسمت دوم
۹	مسئله دوم: قسمت اول
۱۶	مسئله دوم: قسمت دوم
۲۰	مسئله دوم: قسمت سوم و چهارم
۳۳	جمع بندی نتایج

## توضیحات

### توضیحات اولیه، بارگزاری داده ها و پیش پردازش

برای پیاده سازی و پاسخ گویی به این سری از تمرینات از زبان پایتون و کتابخانه tensorflow برای ایجاد و آموزش مدل ها استفاده شده است.

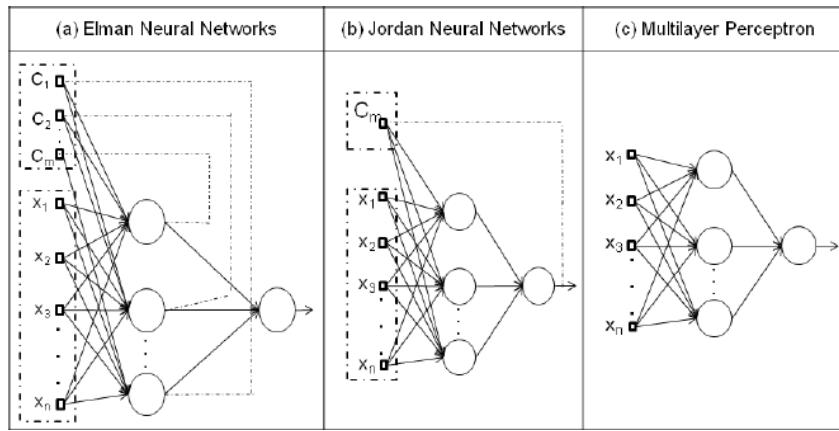
مجموعه داده از لینک داده شده دانلود و بارگزاری آن از `5lp[num].data` فایل با پیشوند `[num]` انجام شده است. در بارگزاری داده ها تمامی فایل ها خوانده شده و در یک رشته ذخیره شدنده؛ سپس با پردازش سطر به سطر آن مجموعه داده عددی تولید و حاصل شده است. هر رکورد از مجموعه داده ای حاصل به شکل (15,6) در آمد که ۱۵ نشانگر گام زمانی و ۶ نیز نشانگر ویژگی های آن گام زمانی می باشد. همچنین برای همگرايی سريع و بهبود روند آموزش شبکه و دور از تاثير گذاري ویژگي هايی که رنج بزرگ دارند، ویژگي ها نرمال شده اند. نرمال سازی انجام شده به ازاي ویژگي های مشابه و در گام های زمان يكسان اعمال شده است.(مثالا همه ویژگي های `in` در گام های زمانی `in` در رکوردهای متفاوت با هم نرمال شده اند.)

اختصاص برچسب نیز بصورت دودوئی اعمال شده و داده هایی که برچسب `ok` یا `normal` داشته اند با برچسب `1` و سایر داده ها با برچسب `0` انکود شده اند. همچنین مجموعه داده به با نسبت های ۲۰، ۷۰ و ۱۰ درصد به گروه های آموزش، آزمون و اعتبار سنجی بصورت تصادفی و طبقه ای تقسیم می شود.

## مسئله اول: قسمت اول

### مسئله اول: قسمت اول

برای پاسخ گویی به این مسئله ابتدا خود شبکه های المن و جردن مورد بررسی و مطالعه قرار گرفته و جنبه های تفاوت آن با MLP آشکار شد:



در شبکه المن، خروجی تمامی نورون های لایه مخفی به عنوان *fully-connected feedback* بصورت *fully-connected feedback* با خود متصل هستند؛ لذا ما به تعداد نورون های لایه مخفی بازخورد برای مرحله بعد دریافت می کنیم. اما در شبکه جردن، خروجی نهایی شبکه به عنوان *feedback* و بصورت *fully-connected* به لایه مخفی متصل می باشد.

با مطالعه انجام شده و طبق توضیحات داکیومنت keras به [لينك](#) متوجه شدم که پیاده سازی انجام شده در آن برای لایه SimpleRNN همان پیاده سازی شبکه المن می باشد که خروجی شبکه مخفی بصورت تماماً متصل به خودش بازخورد داده می شود. لذا برای پیاده سازی شبکه المن از این لایه استفاده شده است. همچنین طبق جست و جو دریافتی که شبکه جردن بصورت پیشفرض پیاده سازی نشده است از این رو بایستی برای آن چاره اندیشیده شود.

برای حل این چالش، بنده از سلول recurrent موجود در keras بصورت شی گرایی ارث بری کرده و با override زمانی `call`، که بایستی مقادیر *feedback* نورون ها بازگردانده شود، لایه softmax روی نتایج اعمال شده و خروجی نهایی که دو نورون می باشد حاصل می شود و آن به عنوان بازخورد برگردانده می شود تا شبکه جردن حاصل شود. اما بایستی توجه داشت باید شکل بازخورد با شکل لایه مخفی حفظ شود؛ مثلا زمانی که ما در لایه مخفی ۲۰ نورون داریم، بایستی دقیقاً ۲۰ بازخورد را فراهم کرده و آن را بازگردانیم در صورتی که وقتی خروجی نهایی مدل را بدست می آوریم صرفاً دو بازخورد(خروجی softmax) در اختیار است و نمیتوان دو بازخورد را به ۲۰ بازخورد بازگرداند؛ از این، در زمانی بازگردان خروجی مدل به عنوان بازخورد به لایه مخفی به تعداد مابه التفاوت صفر پر می شود.(در

این مثال ۱۸-۲۰) این راه حل ارائه و پیاده سازی شده یکی از راهکار های پیاده سازی شبکه جردن می باشد و میتوان از سایر روش ها نظری پیاده سازی یک لایه‌ی جدید و... نیز استفاده نمود:

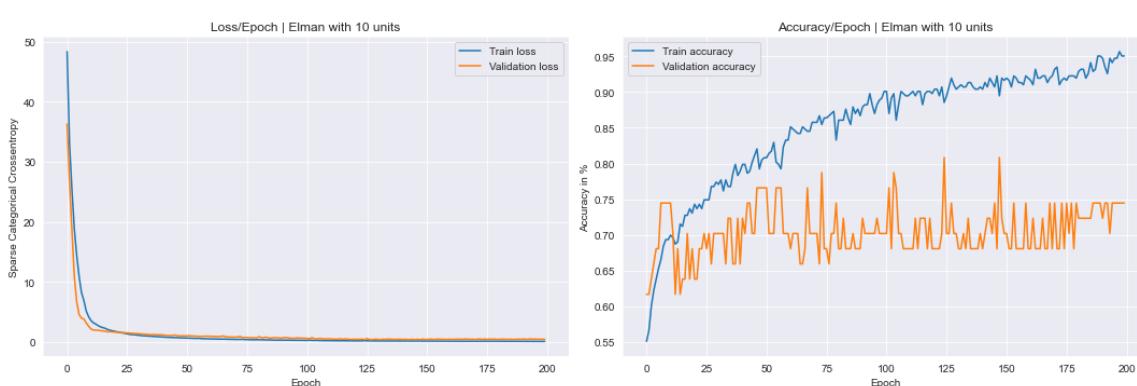
## Jordan Cell

```
class JordanCell(keras.layers.SimpleRNNCell):
    def __init__(self,n=2,**kwargs):
        super(JordanCell, self).__init__(**kwargs)
        self.softmax_layer = tf.keras.layers.Dense(n, activation='softmax')
        self.n = n

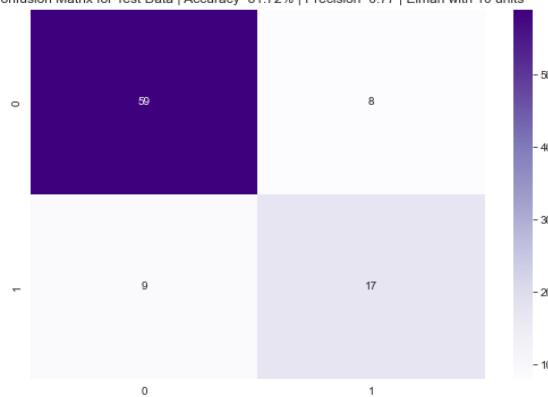
    def call(self, inputs, states, training=None):
        output,new_state = super().call(
            inputs=inputs, states = states, training=training
        )
        new_state = output
        zeros = tf.zeros_like(new_state)
        predict = self.softmax_layer(new_state)
        _merge = Concatenate()([zeros[:, :new_state.shape[1]-self.n],predict])
        return predict,[_merge]
```

هر یک از شبکه های جردن و المن را به ازای سه مقدار ۱۰، ۲۰ و ۵۰ نورون آموزش دادیم که نتایج آن بصورت زیر قابل مشاهده است. همانطورکه قابل ملاحظه می باشد در هر یک از مدل های فوق، با افزایش تعداد نورون های لایه‌ی مخفی، یادگیری داده‌های آموزشی در مدل افزایش یافته و عملکرد مدل در پیش‌بینی داده های unseen در مسیر رو به رشد حرکت می کند که علت آن استخراج ویژگی های بیشتر می باشد و می دانیم پس از یک حد آستانه این باعث بیش برآش می شود.(مقایسه این دو مدل در پاسخگویی به قسمت بعدی) بر حسب تجرب بدهست آمده از تمارین قبلی با چند آموزش آزمایشی، در آموزش های این سری از تمرینات از adam برای optimizer و sparse categorical cross entropy برای محاسبه خطا استفاده شده است. همچنین آموزش های تا ۲۰۰ تکرار ادامه یافته و از ReLU برای تابع فعالیت نورون های لایه‌ی مخفی استفاده شده است. به ازای هر یک آموزش های انجام شده نمودار های loss و accuracy و ماتریس درهم ریختگی(به ازای داده های آزمون) فراهم شده اند. در قسمت title ماتریس درهم ریختگی عملکرد پیش‌بینی نیز درج شده است. همچنین در قسمت title هر یک از نمودار ها، عنوان شبکه‌ی آموزش یافته‌ی مربوطه درج شده است.

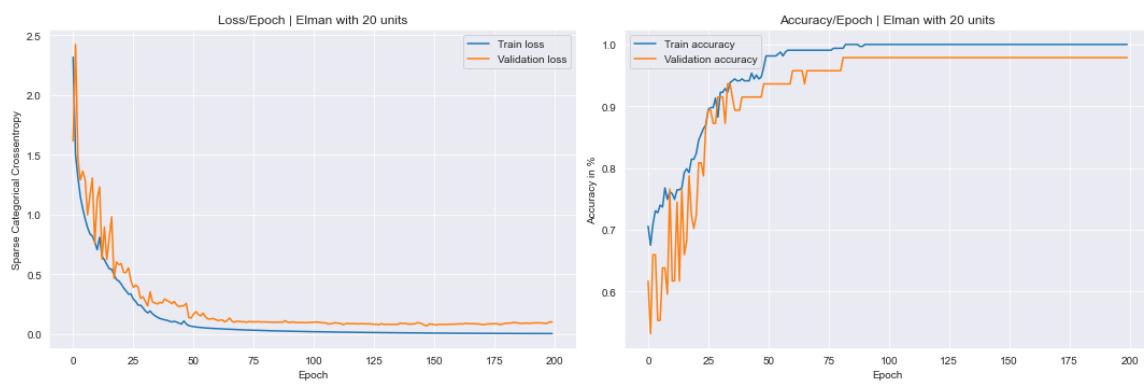
عملکرد شبکه المن به ازای ۱۰ نورون:



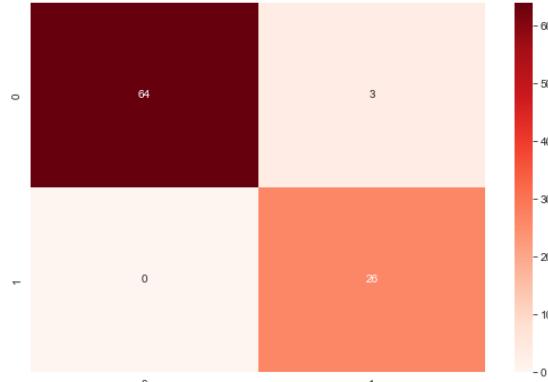
Confusion Matrix for Test Data | Accuracy=81.72% | Precision=0.77 | Elman with 10 units



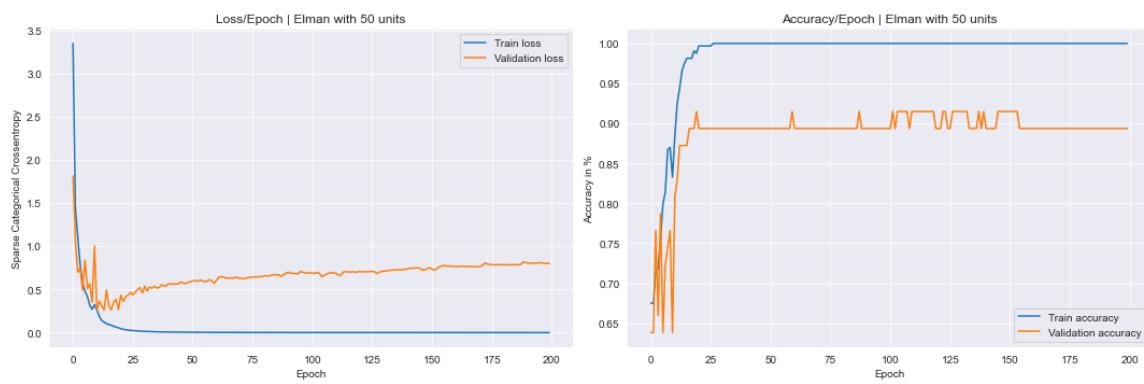
عملکرد شبکه المن به ازای ۲۰ نورون:



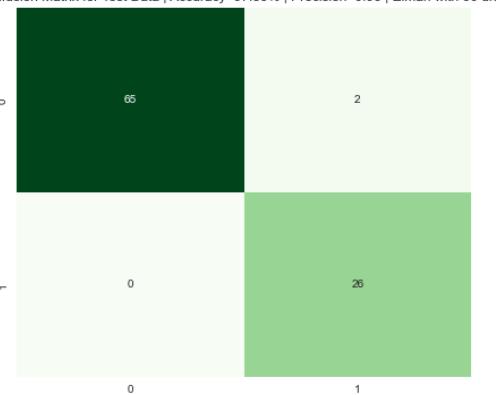
Confusion Matrix for Test Data | Accuracy=96.77% | Precision=0.95 | Elman with 20 units



عملکرد شبکه المن به ازای ۵۰ نورون:

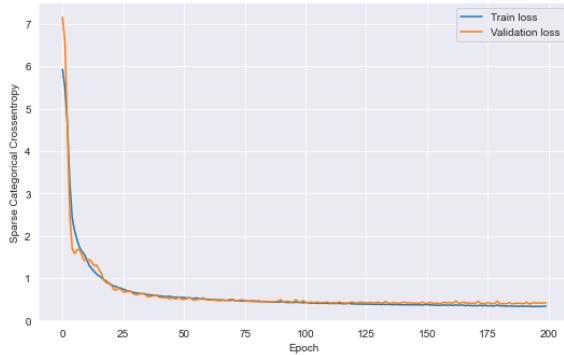


Confusion Matrix for Test Data | Accuracy=97.85% | Precision=0.96 | Elman with 50 units

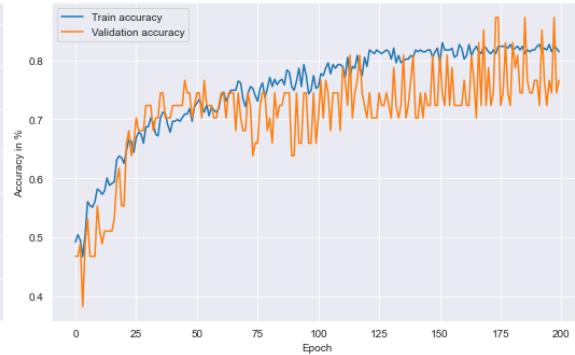


عملکرد شبکه جردن به ازای ۱۰ نورون:

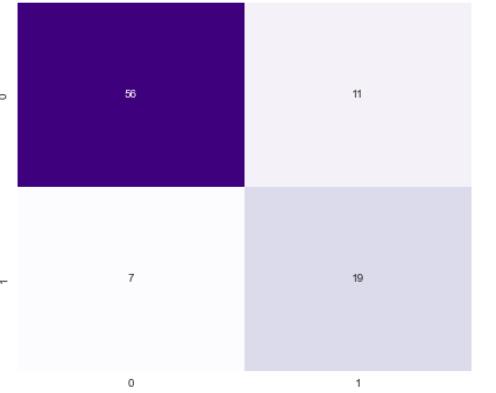
Loss/Epoch | Jordan with 10 units



Accuracy/Epoch | Jordan with 10 units

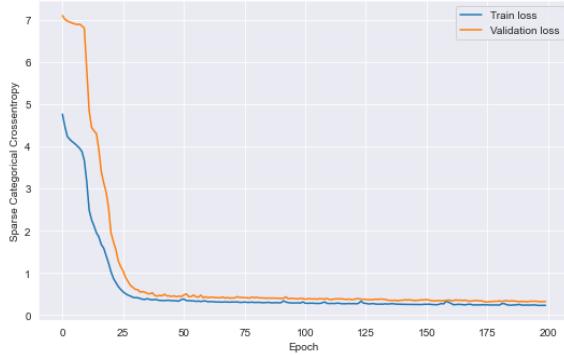


Confusion Matrix for Test Data | Accuracy=80.65% | Precision=0.76 | Jordan with 10 units



عملکرد شبکه جردن به ازای ۲۰ نورون:

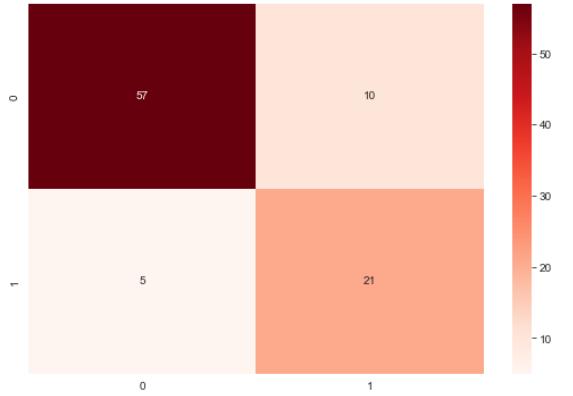
Loss/Epoch | Jordan with 20 units



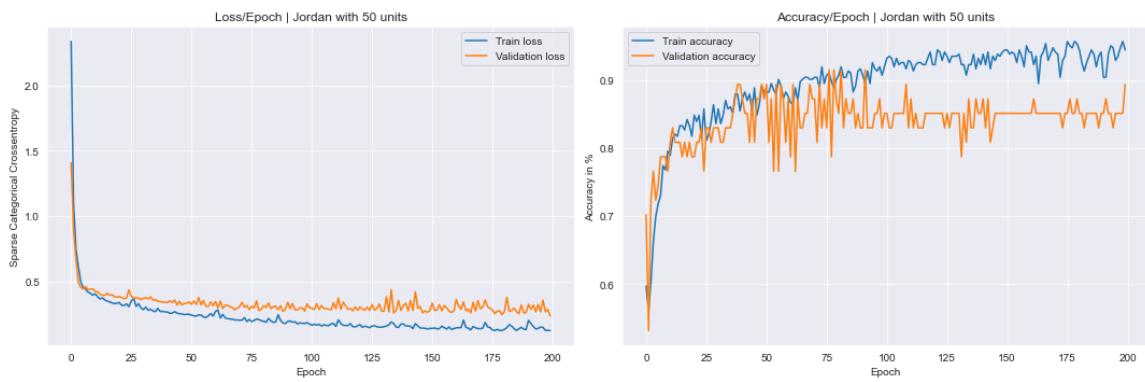
Accuracy/Epoch | Jordan with 20 units



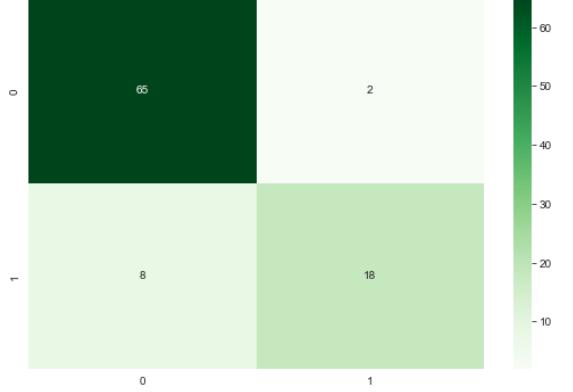
Confusion Matrix for Test Data | Accuracy=83.87% | Precision=0.8 | Jordan with 20 units



عملکرد شبکه جردن به ازای ۵۰ نورون:



Confusion Matrix for Test Data | Accuracy=89.25% | Precision=0.9 | Jordan with 50 units



از نمودار های حاصل از آموزش شبکه های المن می توان دریافت که ۲۰ نورون در لایهی مخفی تعداد نورون بهینه ای محسوب می شود چرا که علاوه بر دقت بالای در پیش بین داده های آزمون، نمودار های loss و accuracy نرمالی داشته و اثری از بیش برازش دیده نمی شود در حالی که به ازای ۵۰ نورون نمودار خطابه سمت بیش برازشی حرکت کرده است.(خود عدد ۲۰ یا اندکی بالاتر از آن مناسب می باشد) همچنین در خصوص شبکه جردن نیز می توان گفت که حداقل ۵۰ نورون در لایهی مخفی تعداد نورون مناسبی محسوب می شود زیرا علاوه بر دقت بالا نسبت به سایر آزمایش های انجام شده، به نظر می رسد نمودار loss آن در شروع راه بیش برازشی می باشد و افزایش تعداد نورون می توان عملکرد مدل را به درجهت بیش برازشی سوق دهد.

## مسئله اول: قسمت دوم

### مسئله اول: قسمت دوم

برای مقایسه‌ی این دو شبکه، نمودار های loss و accuracy را به همراه ماتریس درهم ریختگی و دقت حاصل از پیش‌بینی را توانان مورد مطالعه قرار می‌دهیم. برای هر یک از شبکه‌های مذکور در تمامی آزمایش‌های انجام شده به ازای تعداد نورون‌های یکسان، عملکرد مدل‌المن بهتر از عملکرد مدل جردن می‌باشد؛ در حالت ۱۰ نورونی این اختلاف تقریباً ۲ درصد، در حالت ۲۰ نورونی تقریباً ۱۳ درصد و در حالت ۵۰ نورونی نیز تقریباً ۸ درصد می‌باشد. مبنای مقایسه‌ی عملکرد شامل دقت پیش‌بینی داده‌های unseen می‌باشد.

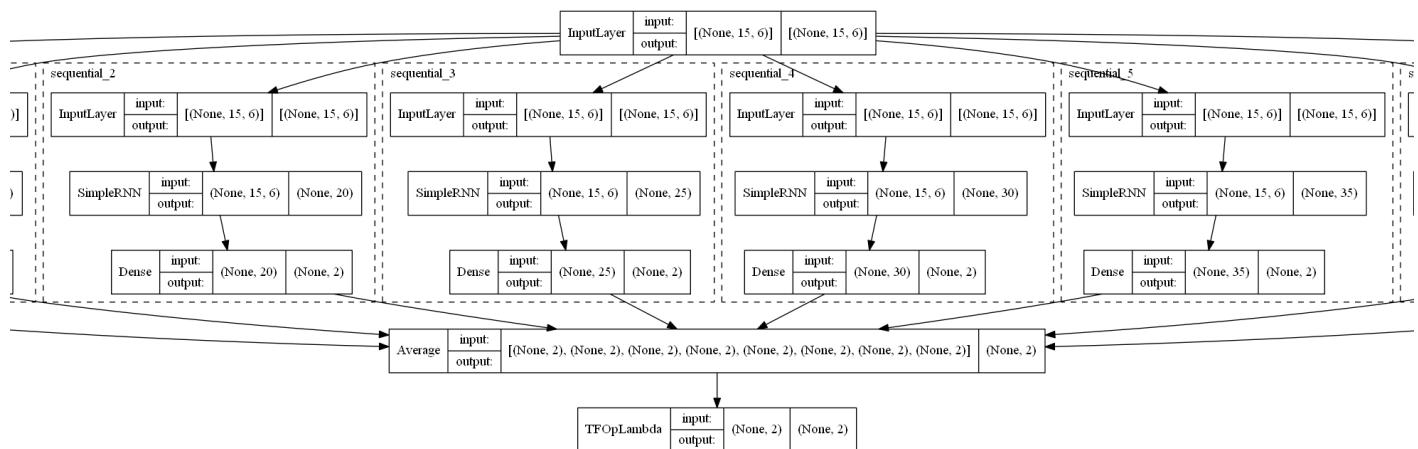
همانطور که در قسمت قبل ذکر شده است، شبکه‌المن از تمامی نورون‌های خود بصورت fully-connected بازخورد دریافت می‌کند در حالی که شبکه جردن صرفاً از خروجی نهایی مدل باز خورد دریافت می‌کند لذا با افزایش تعداد نورون لایه‌ی مخفی در شبکه‌المن، باز خورد های وارد به لایه‌ی مخفی نیز افزایش یافته و به شبکه برای یادگیری بهتر داده‌ها کمک می‌کند و این در حالی است که در شبکه جردن و در تمامی حالت‌ها صرفاً دو بازخورد (مسئله مان دودوئی است) به لایه‌ی مخفی بازخورد بازگردانده شده و مستقل از تعداد نورون‌های لایه‌ی مخفی می‌باشد. از این رو و بر اساس نتایج حاصل از آموزش میتوان انتظار آن را داشت که با افزایش تعداد نورون در لایه‌ی مخفی، شبکه‌المن بتواند آگاهی بیشتری از گام‌های قبلی داشته و بهتر آموزش بینید و عملکرد بهتری نیز داشته باشد.

بر اساس نتایج تجربی حاصل می‌توان گفت شبکه‌المن انتخابی خوبی برای آموزش این مجموعه داده می‌باشد چرا که در تعداد نورون‌های مشابه عملکرد بهتری از جردن داشته است؛ بر اساس تئوری نیز از جایی که به ازای هر حرکت تعداد گام‌های زمانی بیشتر از تعداد ویژگی‌های هر گام است بایستی آگاهی بیشتر از گام‌های قبلی به مدل بازخورد داده شود تا بتواند عملکرد خوبی از خود نشان داده و دقیق‌تر آموزش بینید لذا شبکه‌المن که می‌تواند تعداد بازخورد های بیشتری را منتقل کند انتخاب بهتری است.

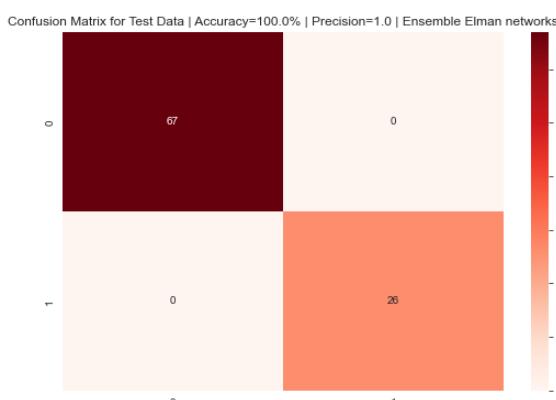
## مسئله دوم: قسمت اول

### مسئله دوم: قسمت اول

برای پاسخ گویی به این بخش، به ازای هر یک از شبکه های المن و جردن یک شبکه گروهی(Ensemble) ایجاد می نماییم. جزئیات و پارامتر های آموزشی مانند سوال قبل در نظر گرفته است. برای هر یک از شبکه ها یک ساختار گروهی ۸ تایی ایجاد می کنیم که در آن معماری هر زیر شبکه اندکی با دیگری متفاوت است. معماری زیر شبکه های انتخابی شامل شبکه های المن با لایه های مخفی با تعداد نورون های ۱۰، ۱۵، ۲۰، ۲۵، ۳۰، ۳۵، ۴۰ و ۴۵ می باشد.(برای جردن نیز از ۵۰ تا ۸۵) برای ایجاد Ensemble، هر یک از هشت شبکه را بصورت موازی و مجزا از هم با همه داده ها آموزش داده و سپس در یک مدل دیگر خروجی آنها را باهم ادغام می کنیم(میانگین خروجی شبکه ها) و سپس با اعمال تابع فعالیت softmax نتیجه نهایی را حاصل می کنیم. تصویر معماری ساختار های گروهی مورد استفاده در پوشه ضمیمه شده قابل مشاهده بوده و در زیر نیز بررسی از آن قابل ملاحظه است:



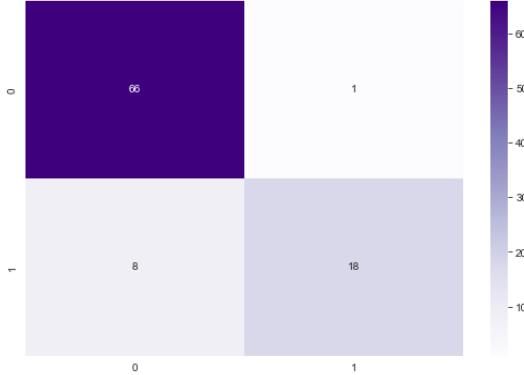
نتیجه مدل Ensemble شده با زیر شبکه های المن:



Passed: 50  
 3/3 [=====] - 0s 3ms/step - loss: 0.2477 - accuracy: 0.8925  
 Passed: 55  
 3/3 [=====] - 0s 4ms/step - loss: 0.2520 - accuracy: 0.8602  
 Passed: 60  
 3/3 [=====] - 0s 5ms/step - loss: 0.3152 - accuracy: 0.8817  
 Passed: 65  
 3/3 [=====] - 0s 6ms/step - loss: 0.2291 - accuracy: 0.8817  
 Passed: 70  
 3/3 [=====] - 0s 4ms/step - loss: 0.2215 - accuracy: 0.8925  
 Passed: 75  
 3/3 [=====] - 0s 5ms/step - loss: 0.1931 - accuracy: 0.9140  
 Passed: 80  
 3/3 [=====] - 0s 5ms/step - loss: 0.2073 - accuracy: 0.9032  
 Passed: 85  
 3/3 [=====] - 0s 4ms/step - loss: 0.2438 - accuracy: 0.8817

نتیجه مدل Ensemble شده با زیر شبکه های جردن:

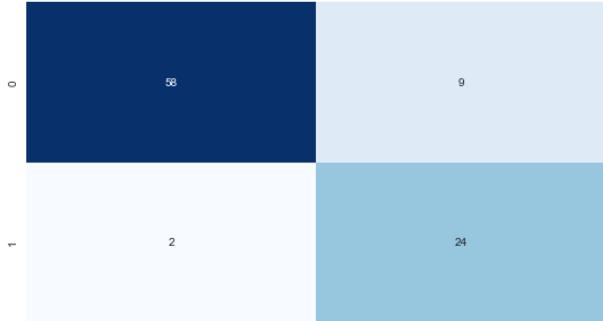
Confusion Matrix for Test Data | Accuracy=90.32% | Precision=0.92 | Ensemble Jordan networks



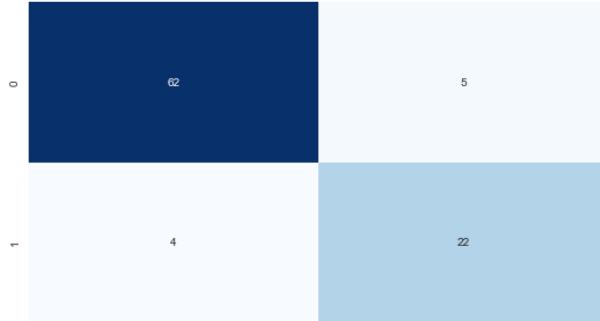
```
Passed: 10  
3/3 [=====] - 0s 2ms/step - loss: 0.1977 - accuracy: 0.8817  
Passed: 15  
3/3 [=====] - 0s 4ms/step - loss: 0.2486 - accuracy: 0.9032  
Passed: 20  
3/3 [=====] - 0s 4ms/step - loss: 0.0138 - accuracy: 1.0000  
Passed: 25  
3/3 [=====] - 0s 2ms/step - loss: 0.0452 - accuracy: 0.9785  
Passed: 30  
3/3 [=====] - 0s 4ms/step - loss: 0.1076 - accuracy: 0.9677  
Passed: 35  
3/3 [=====] - 0s 4ms/step - loss: 2.8765e-04 - accuracy: 1.0000  
Passed: 40  
3/3 [=====] - 0s 4ms/step - loss: 0.0034 - accuracy: 1.0000  
Passed: 45  
3/3 [=====] - 0s 3ms/step - loss: 0.1075 - accuracy: 0.9785
```

## نمودار های ماتریس در هم ریختگی زیر شبکه های آموزش دیده شده ایمن:

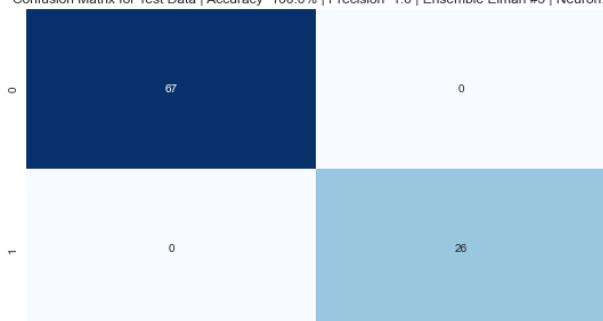
Confusion Matrix for Test Data | Accuracy=88.17% | Precision=0.85 | Ensemble Elman #1 | Neuron:10



Confusion Matrix for Test Data | Accuracy=90.32% | Precision=0.88 | Ensemble Elman #2 | Neuron:15



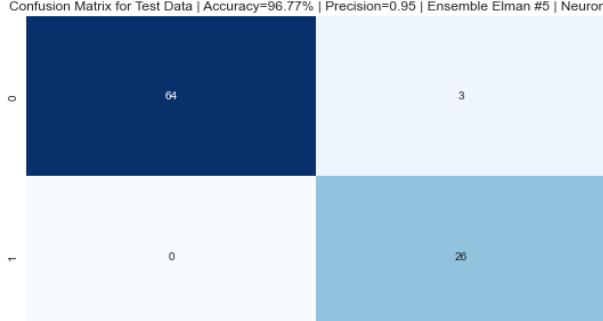
Confusion Matrix for Test Data | Accuracy=100.0% | Precision=1.0 | Ensemble Elman #3 | Neuron:20



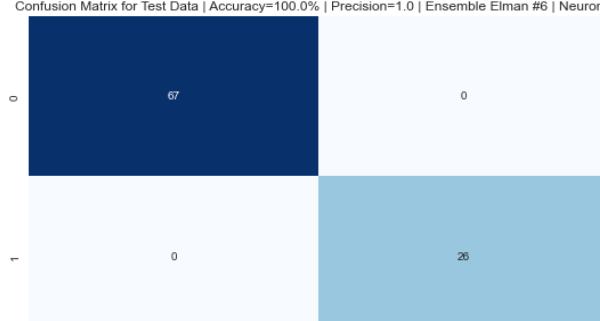
Confusion Matrix for Test Data | Accuracy=97.85% | Precision=0.96 | Ensemble Elman #4 | Neuron:25



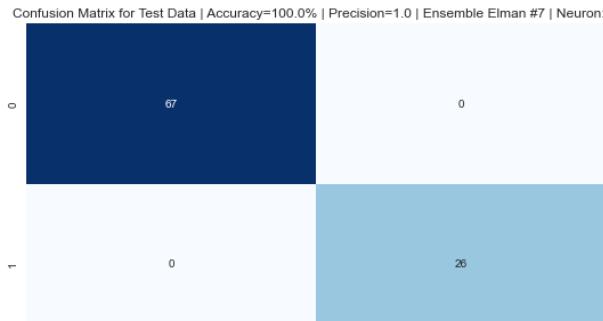
Confusion Matrix for Test Data | Accuracy=96.77% | Precision=0.95 | Ensemble Elman #5 | Neuron:30



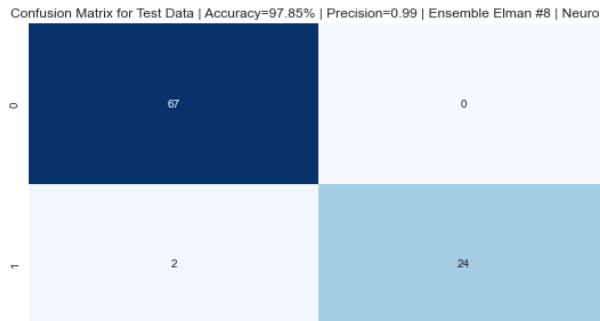
Confusion Matrix for Test Data | Accuracy=100.0% | Precision=1.0 | Ensemble Elman #6 | Neuron:35



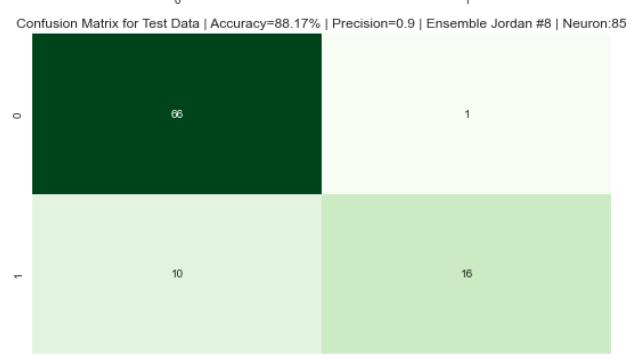
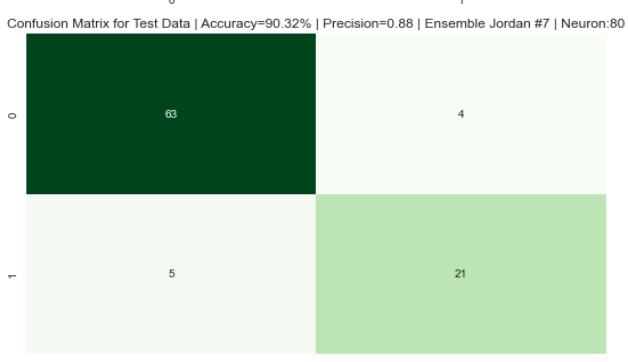
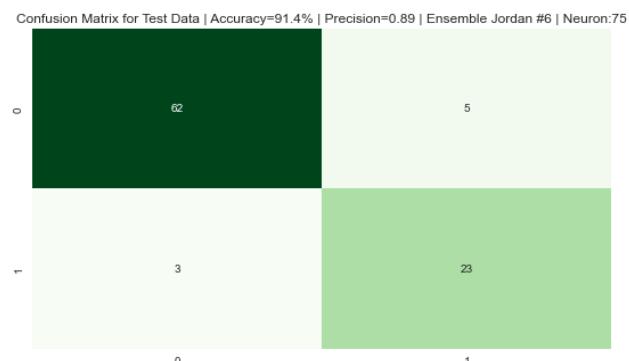
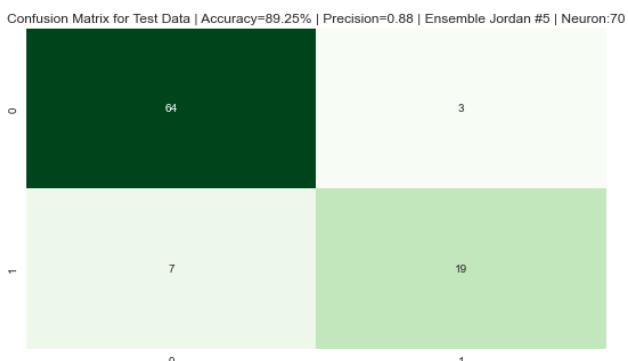
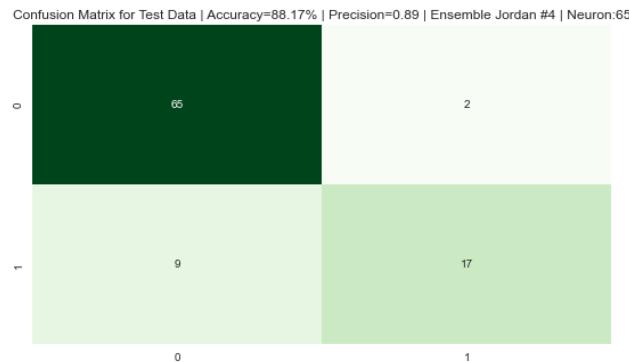
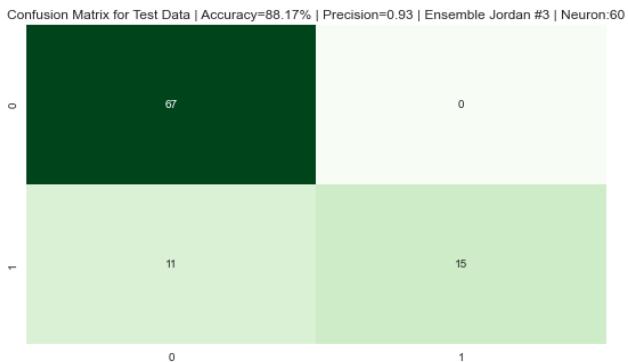
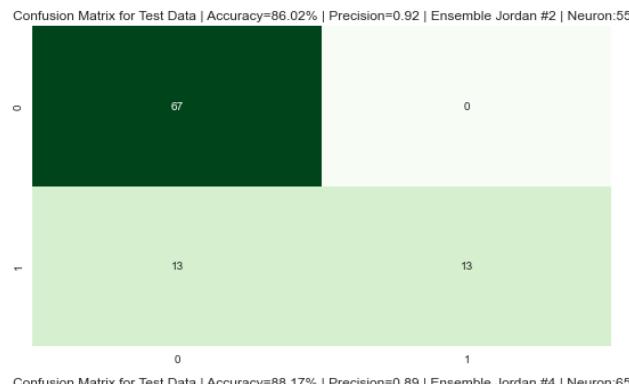
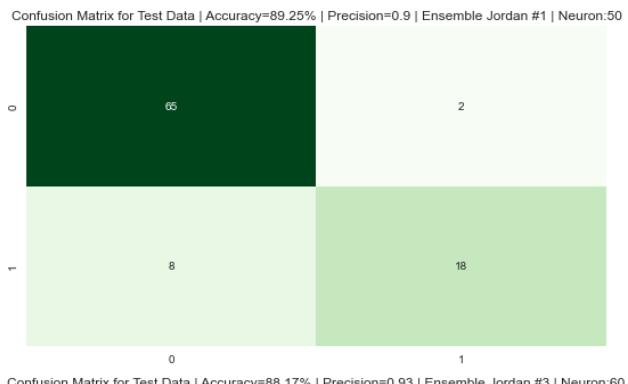
Confusion Matrix for Test Data | Accuracy=100.0% | Precision=1.0 | Ensemble Elman #7 | Neuron:40



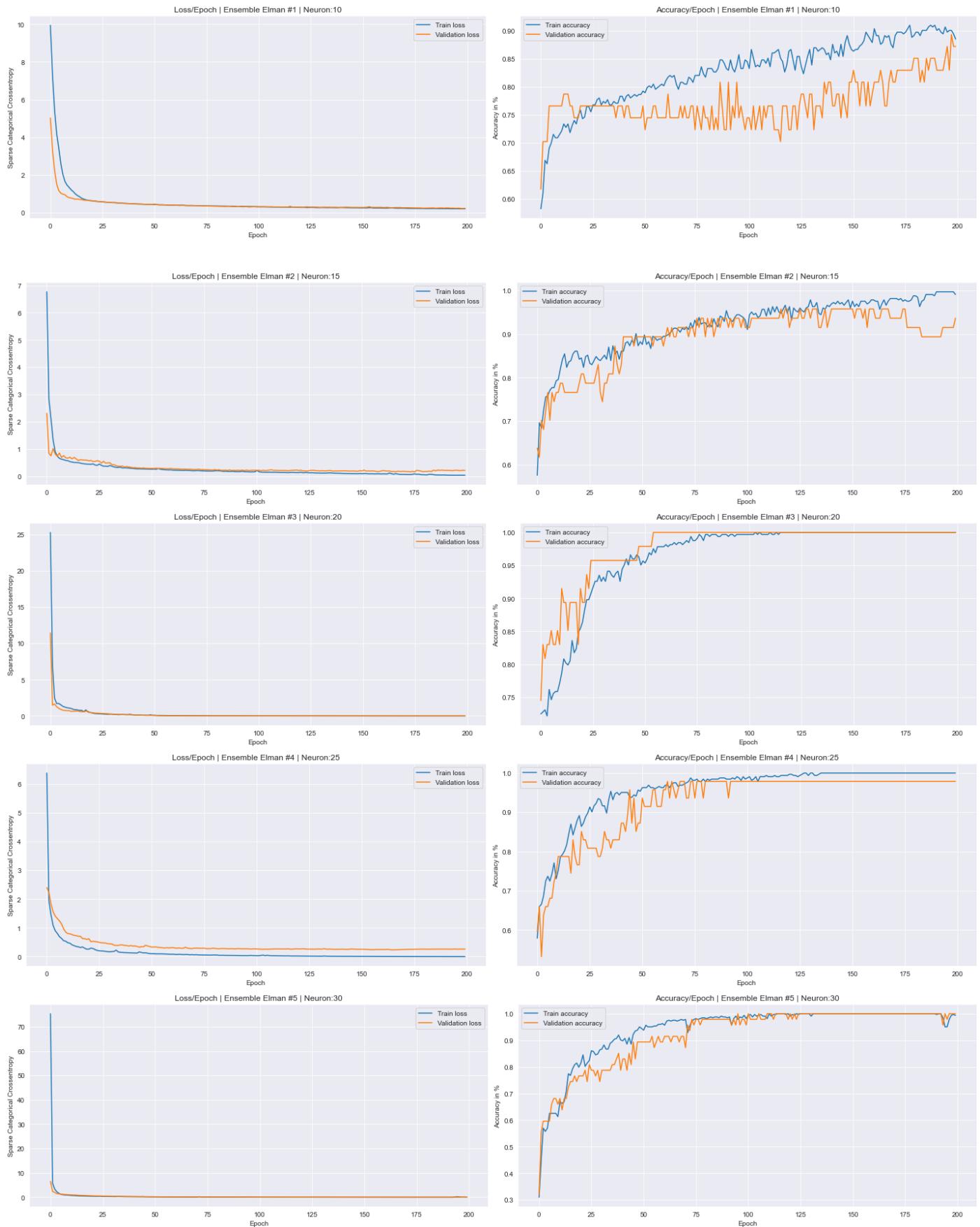
Confusion Matrix for Test Data | Accuracy=97.85% | Precision=0.99 | Ensemble Elman #8 | Neuron:45

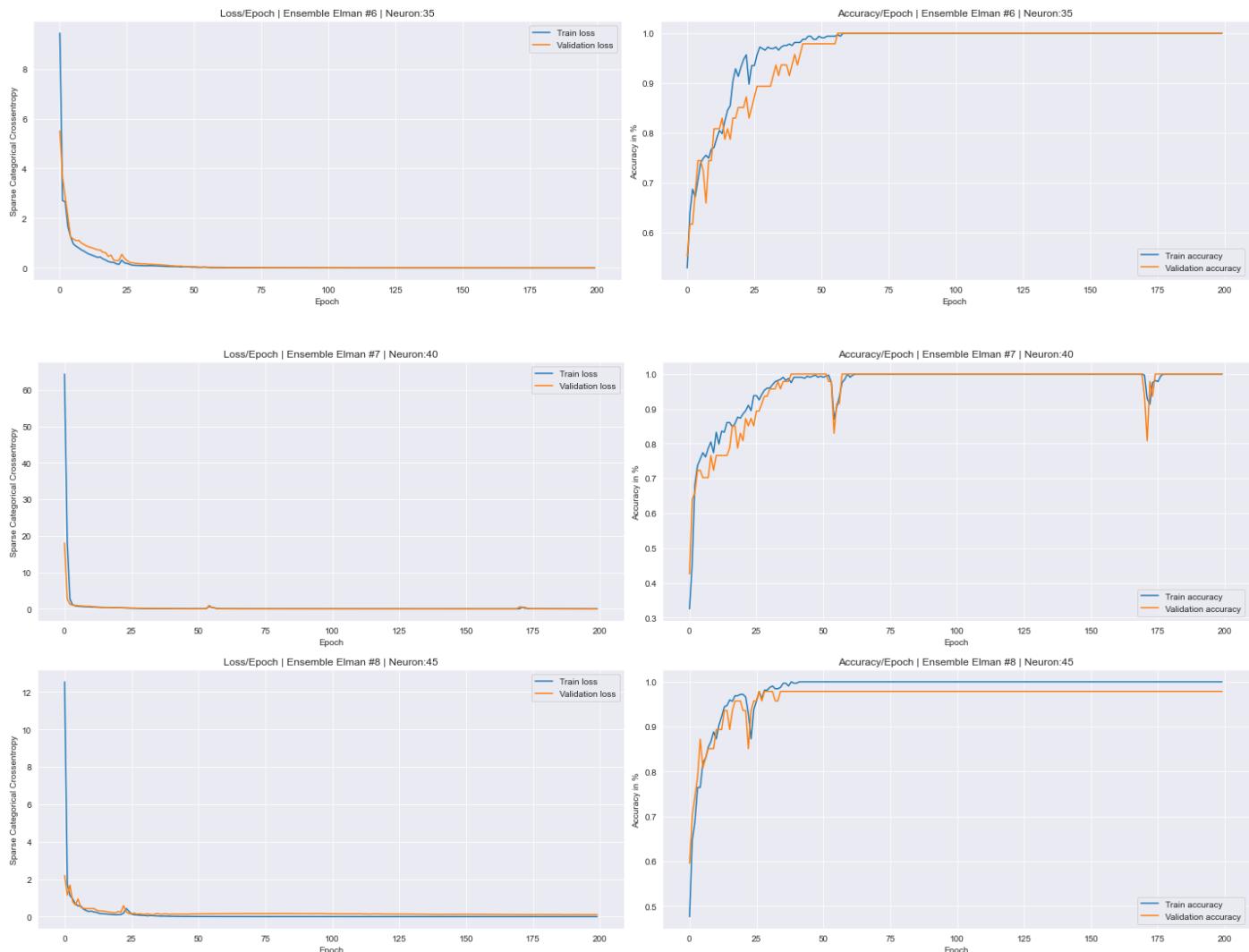


## نمودار های ماتریس در هم ریختگی زیرشبکه های آموزش دیده شده جردن:

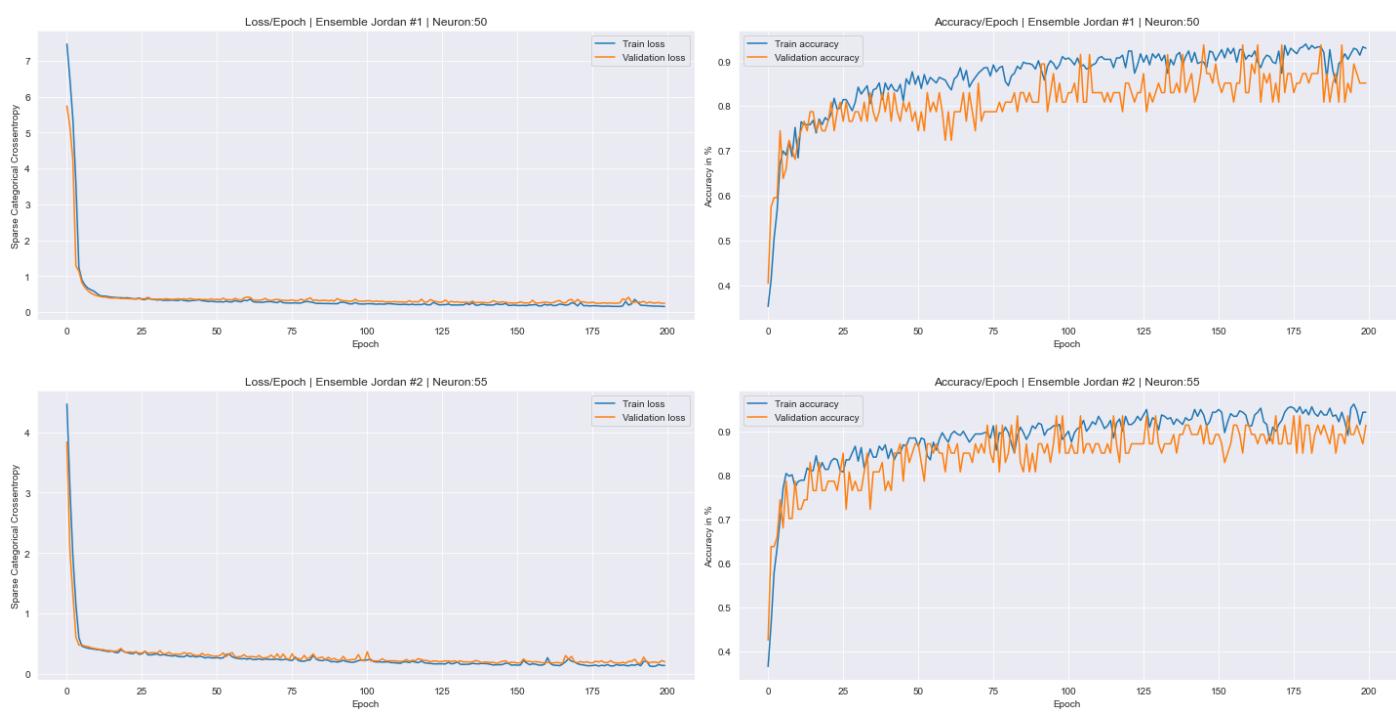


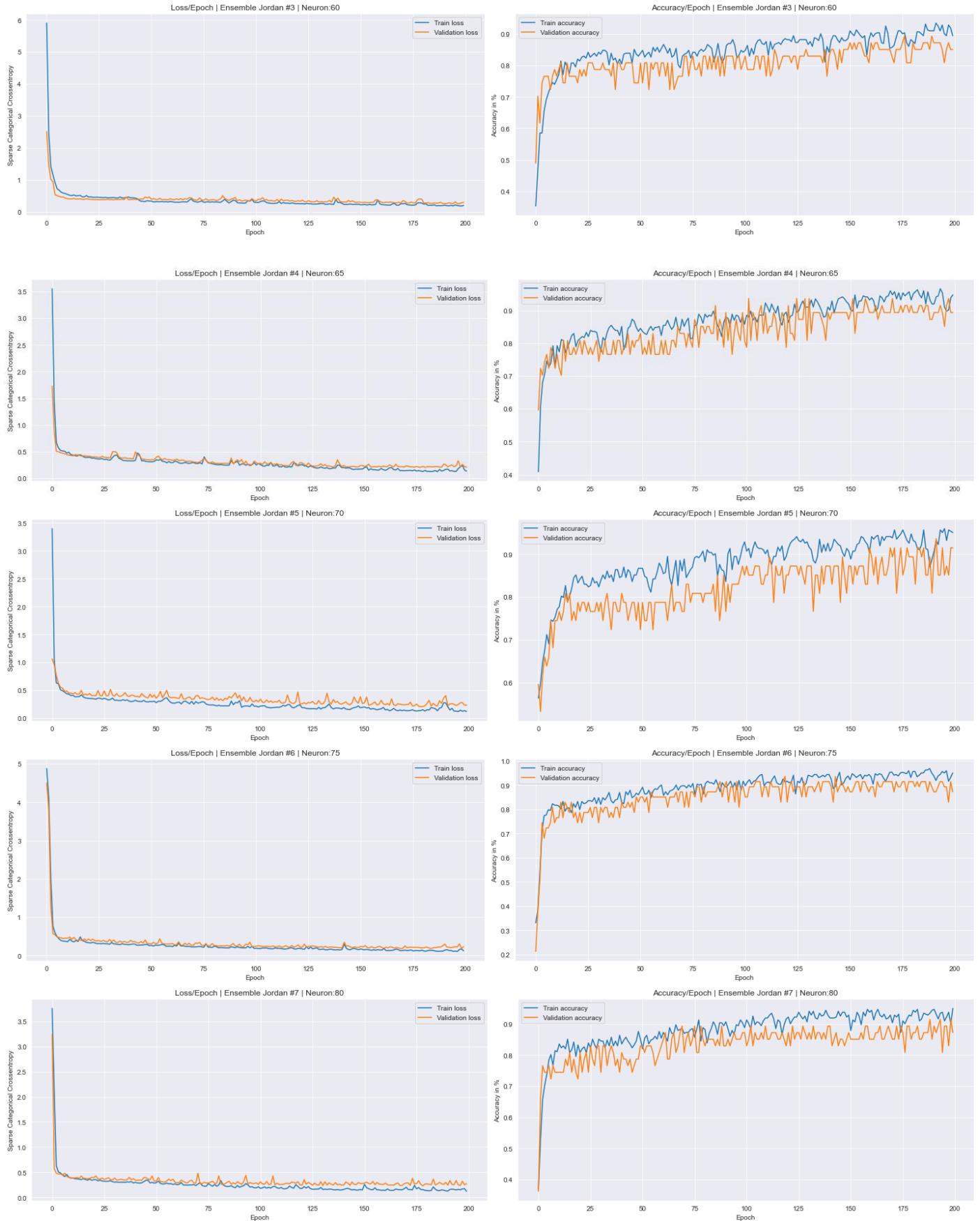
## نمودار های accuracy و loss زیرشبکه های آموزش دیده شدهی المن:

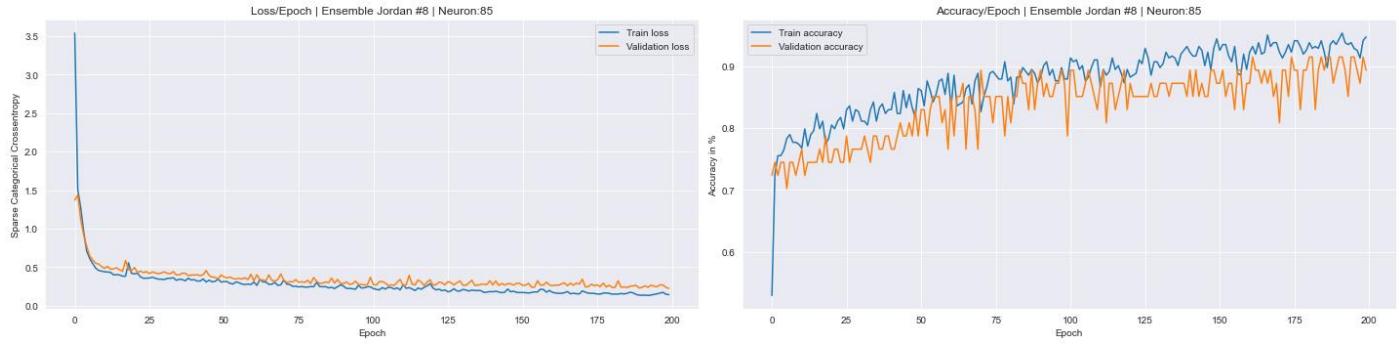




نمودار های accuracy و loss زیر شبکه های آموزش دیده شده جردن:







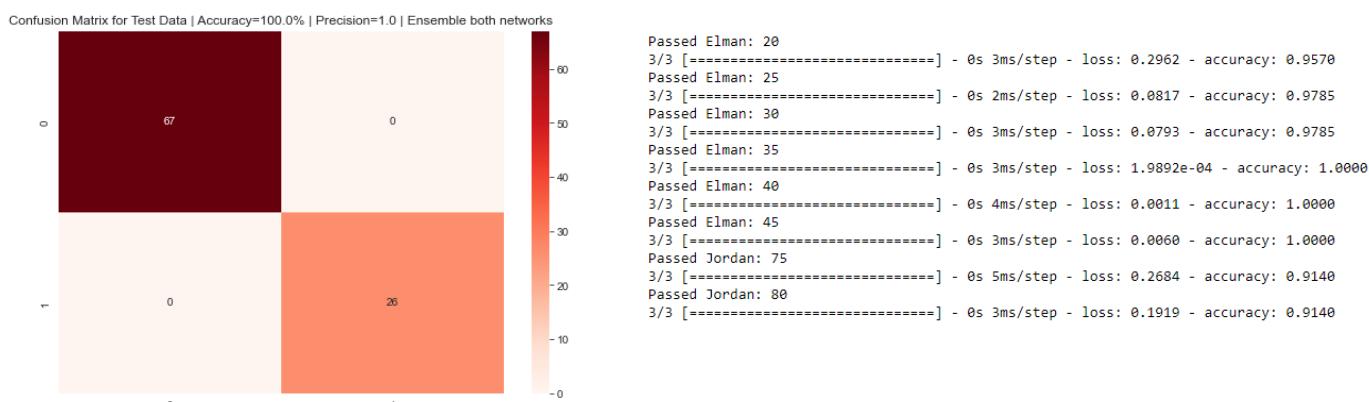
عملکرد حاصل از Ensemble کردن زیر شبکه های المن بسیار بهتر از جردن بوده و دقیق ترین حالت ممکن با دقت پیش‌بینی ۱۰۰٪ی باشد. نمودار های loss و accuracy زیر شبکه های آن نیز نشان میدهد که هیچ یک دچار بیش برآش نبوده است. البته این انتظار وجود داشت که Ensemble کردن با زیر شبکه های المن نتیجه بهتری را حاصل کند چرا که ترکیب شبکه های قدرتمند و میانگین گیری از پیش‌بینی معتبر آنها، نتیجه بهتری را نسبت به ترکیب شبکه های ضعیف‌تر با پیش‌بینی ضعیف‌تر نتیجه می‌دهد.

## مسئله دوم: قسمت دوم

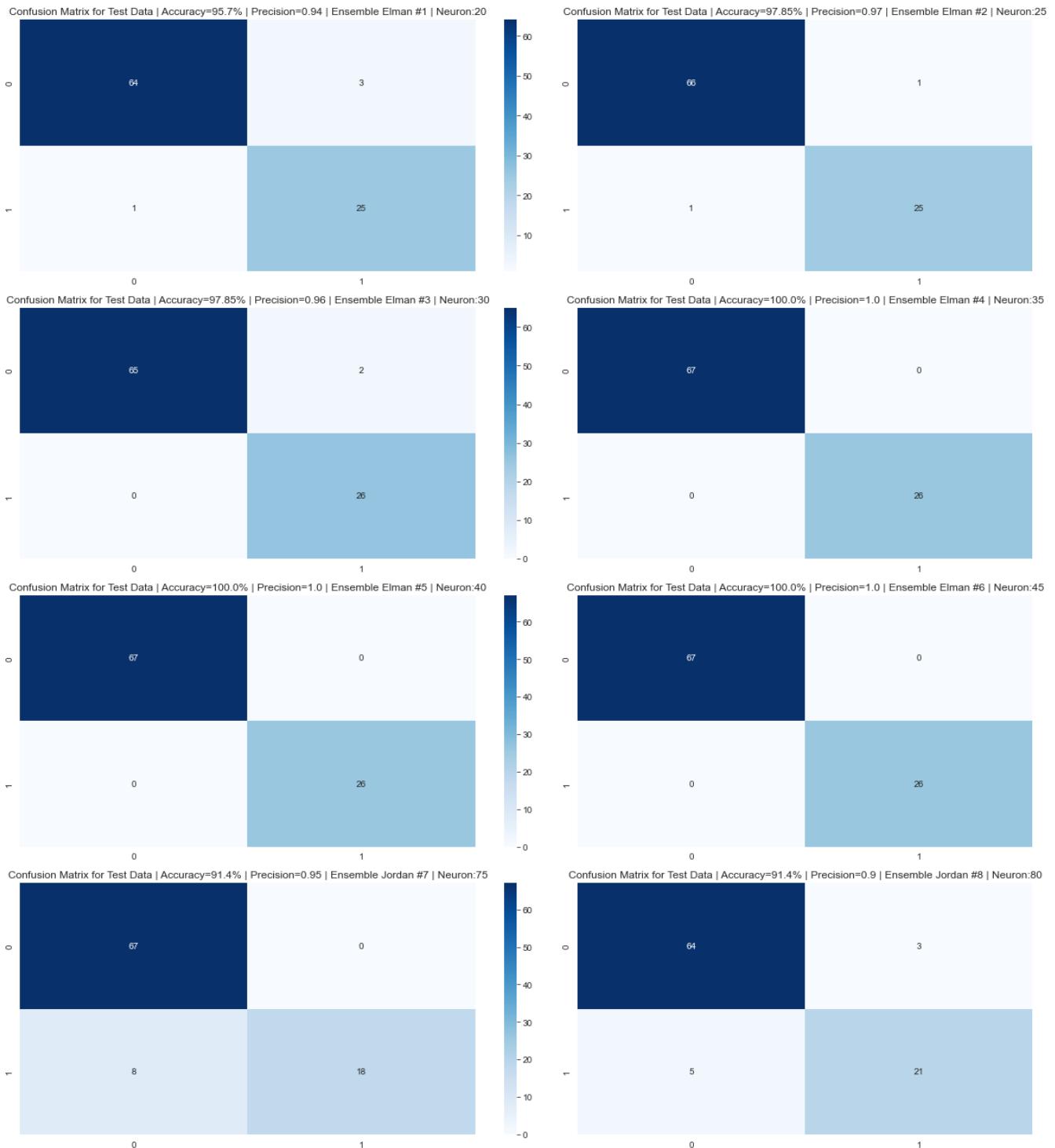
### مسئله دوم: قسمت دوم

در طی آزمایش و آموزش های انجام شده در قسمت قبل توانستیم بهترین ساختار ممکن برای دسته‌بندی داده‌ها را یافته و به دقت ۱۰۰٪ برای پیش‌بینی داده‌های unseen برسیم؛ با این وجود و بر حسب خواسته سوال، چند شبکه‌من و جردن را آموزش داده و با Ensemble کردن آنها اقدام به دسته‌بندی داده‌ها می‌نماییم.

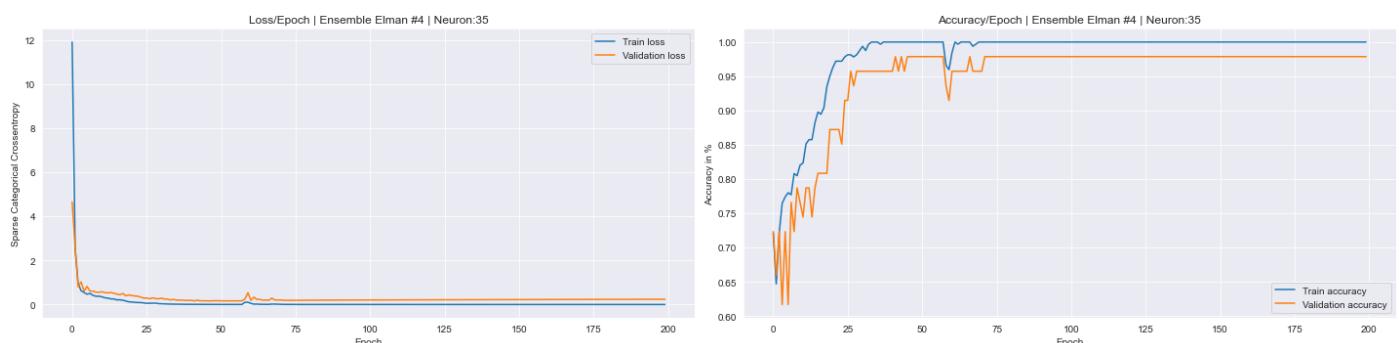
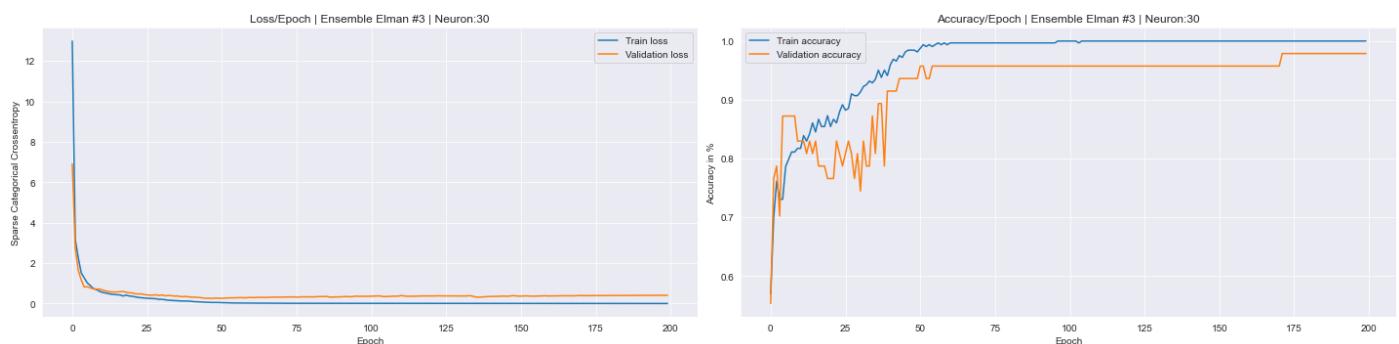
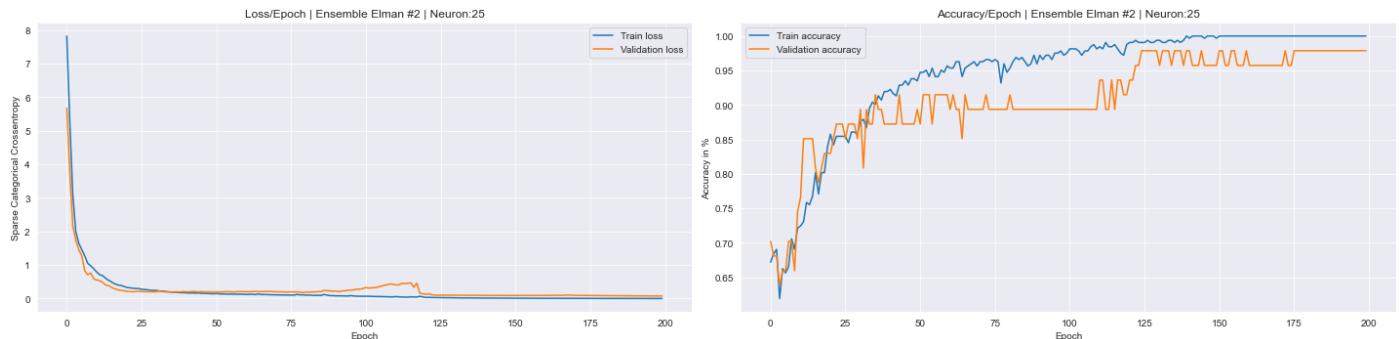
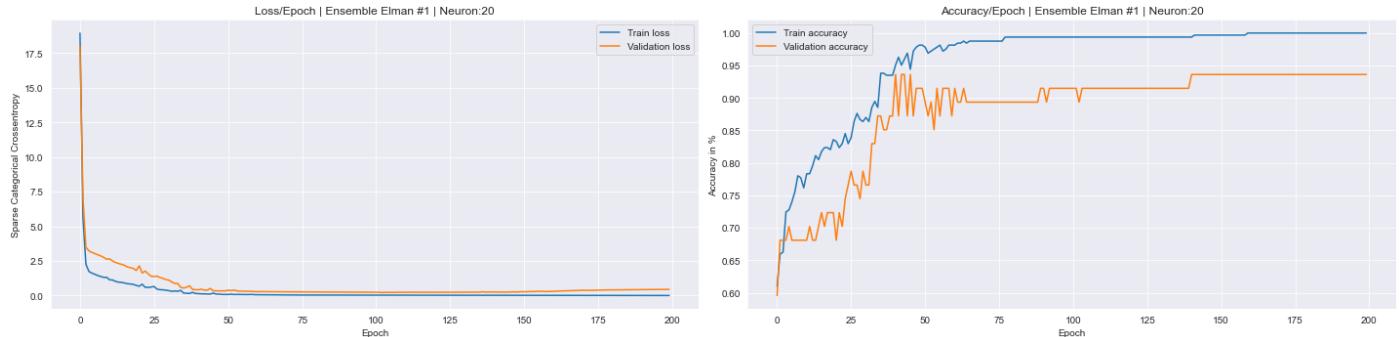
برای انتخاب زیر شبکه‌های مناسب از خروجی قسمت قبل استفاده می‌کنیم؛ ملاحظه کردیم که زیر شبکه‌های المنی با تعداد نورون‌های ۲۰، ۲۵، ۳۰، ۳۵، ۴۰ و ۴۵ بهترین نتیجه را داشته‌اند و از طرفی زیر شبکه‌های جردنی با تعداد نورن ۷۵ و ۸۰ مطلوب‌ترین عملکرد را داشته‌اند؛ از این‌رو، هشت معماری فوق را مجدد ایجاد و آموزش داده و در نهایت آنها را Ensemble کرده و اقدام به دسته‌بندی داده‌های unseen می‌کنیم. نتایج بصورت زیر حاصل می‌شود؛ همانطور که انتظار می‌رود این ساختار گروهی که حاصل ترکیب قدرتمندترین زیر شبکه‌ها می‌باشد، توانسته است مطلوب‌ترین نتیجه یعنی دقت ۱۰۰ درصدی را به ارمغان آورد. (عملکرد هیچ کدام از زیر شبکه‌ها بیش برآش نشده‌اند) در قسمت title هر یک توضیحات مورد نیاز درج شده است:

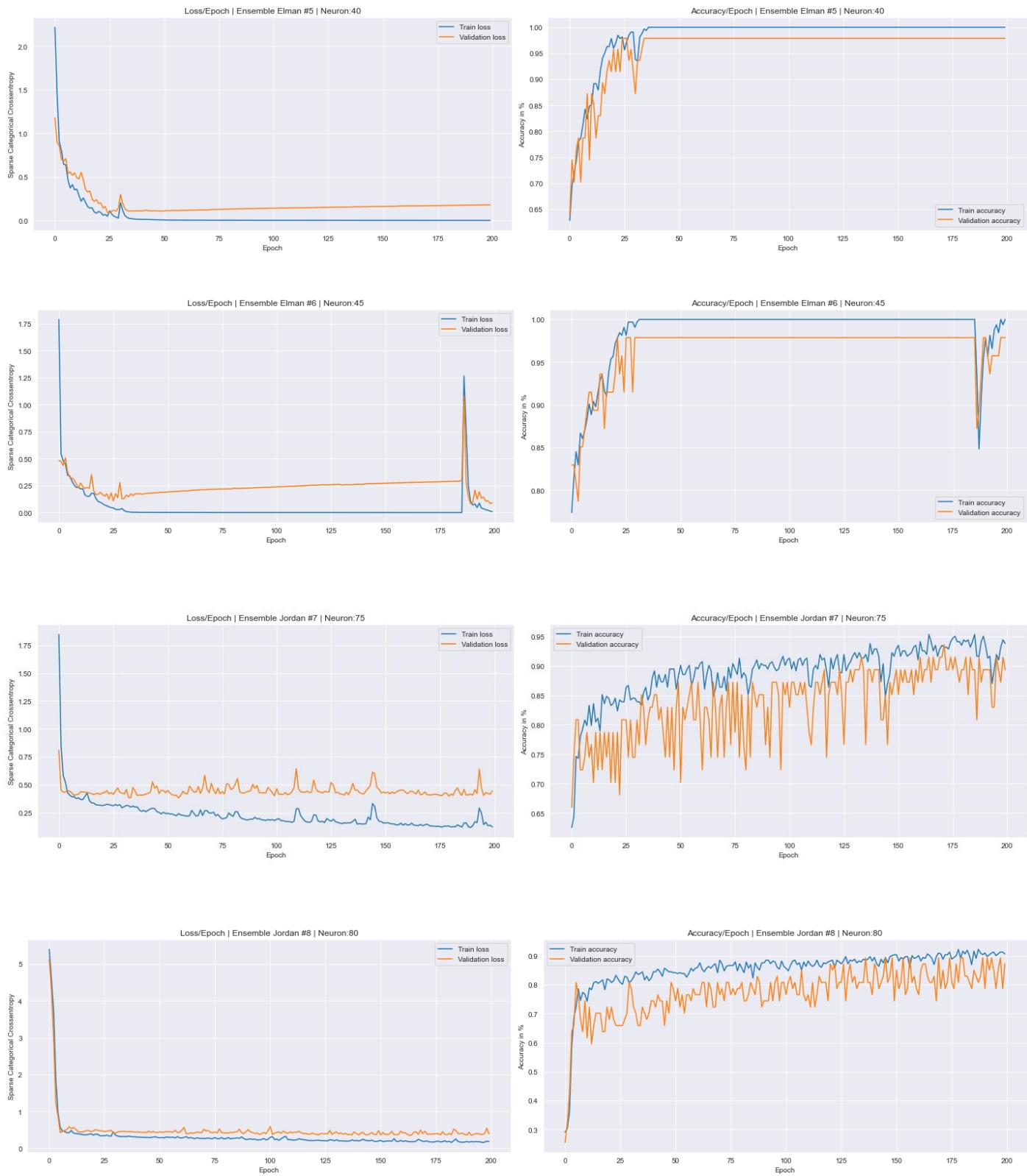


نمودارهای ماتریس در هم ریختگی زیر شبکه‌های ادغامی:  
(صفحه بعدی)



نمودار های loss و accuracy زیر شبکه های ادغامی:



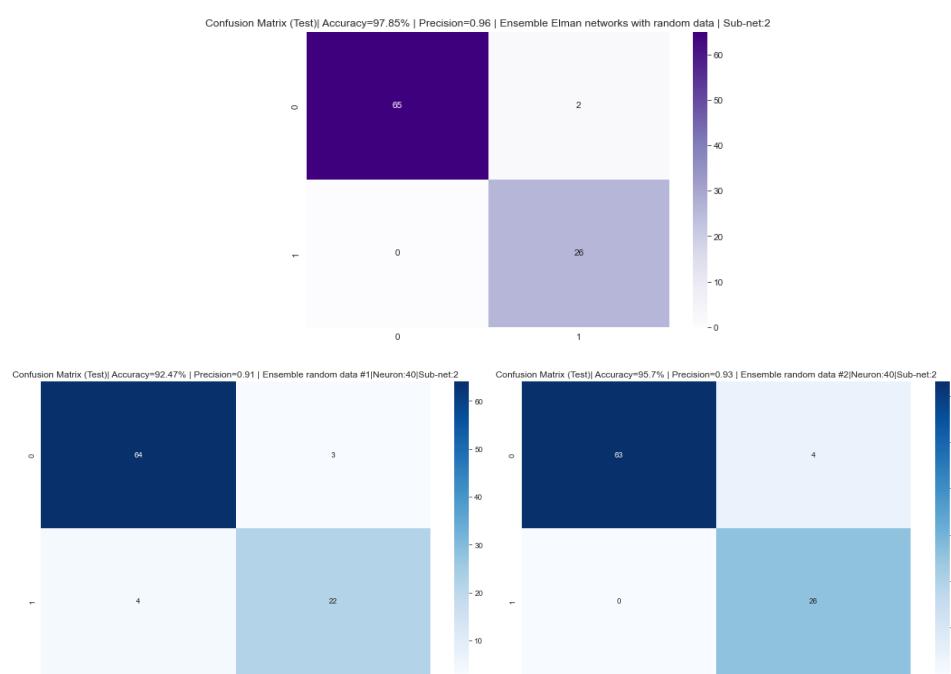


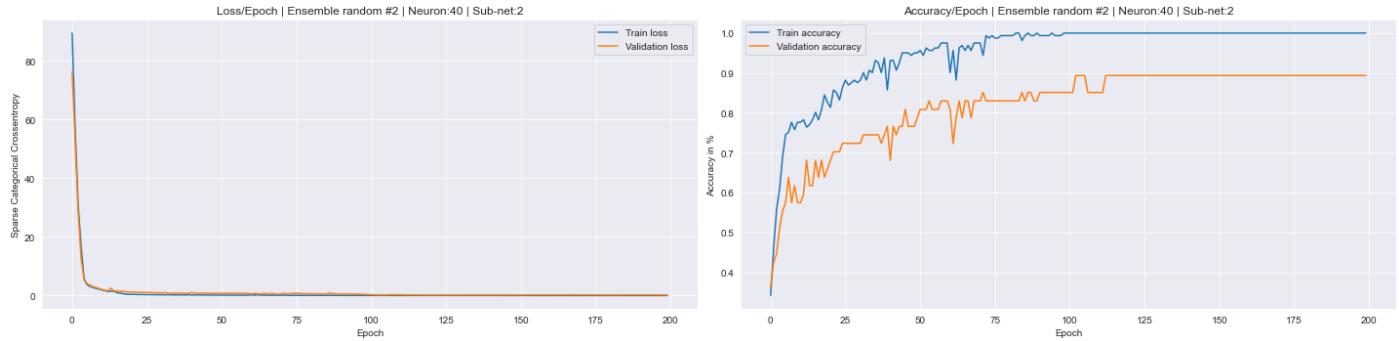
## مسئله دوم: قسمت سوم و چهارم

### مسئله دوم: قسمت سوم و چهارم

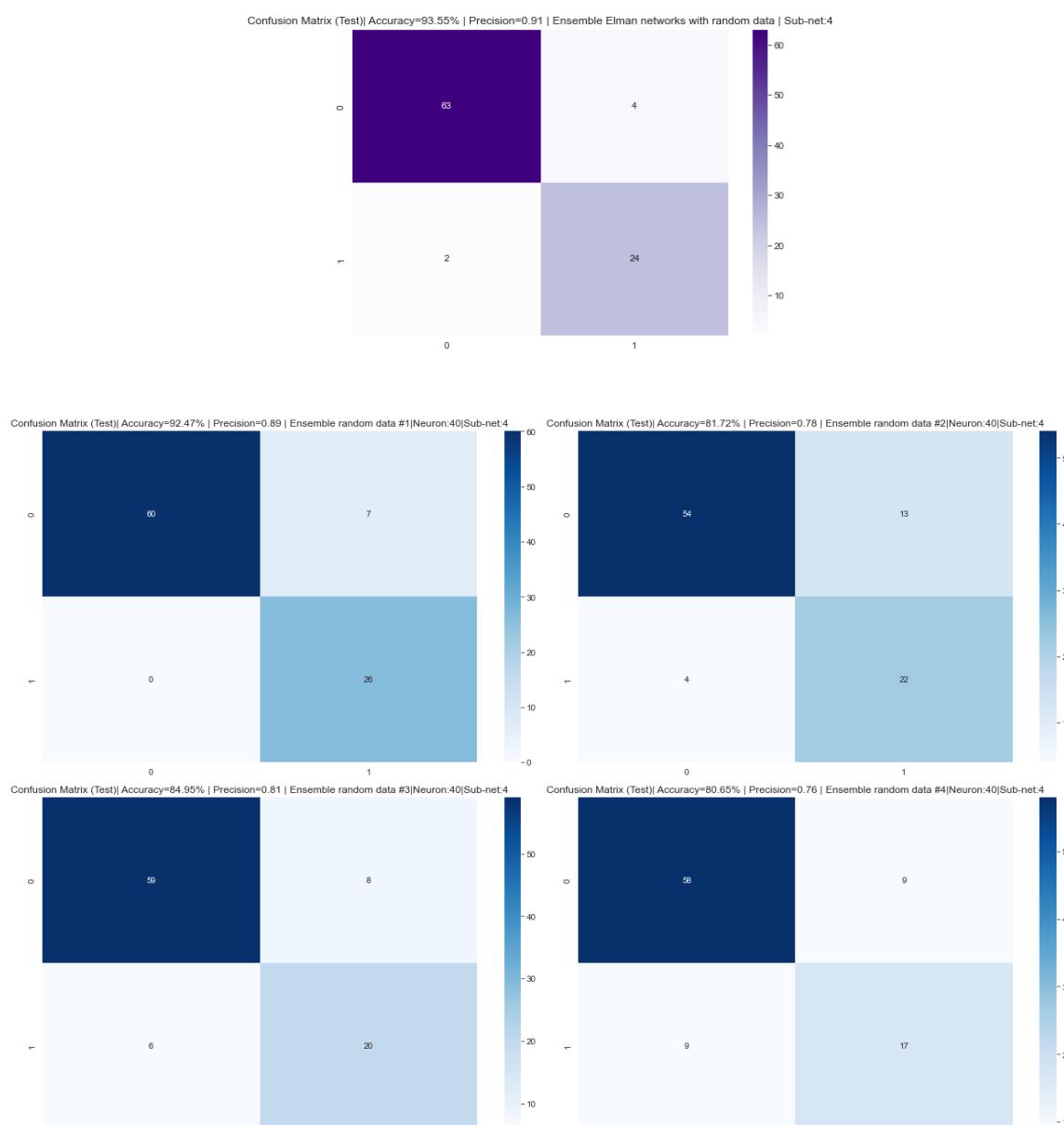
برای پاسخ به این پرسش، یکتابع کمکی به نام Separate پیاده سازی شده است که تعداد دسته های تصادفی مدنظر را گرفته و اندیس های انتخابی برای هر گروه را برابر می گرداند به طوری که با یکدیگر همپوشانی نداشته باشند و از هر دو کلاس به یک نسبت حضور داشته باشند. با توجه به اینکه در متن این دو پرسش گفته نشده از که از چه معماری استفاده شود، لذا بنده از زیر شبکه های المن و هر یک با ۴۰ نورون (که در آزمایش های قبل بهترین نتیجه را داشته اند) استفاده میکنم. دلیل یکسان در نظر گرفتن معماری های زیرشبکه ها نیز ایجاد بستر مناسب برای مطالعه تاثیر تعداد زیر شبکه ها در حالت پخش تصادفی داده های آموزشی بین زیر شبکه ها می باشد.

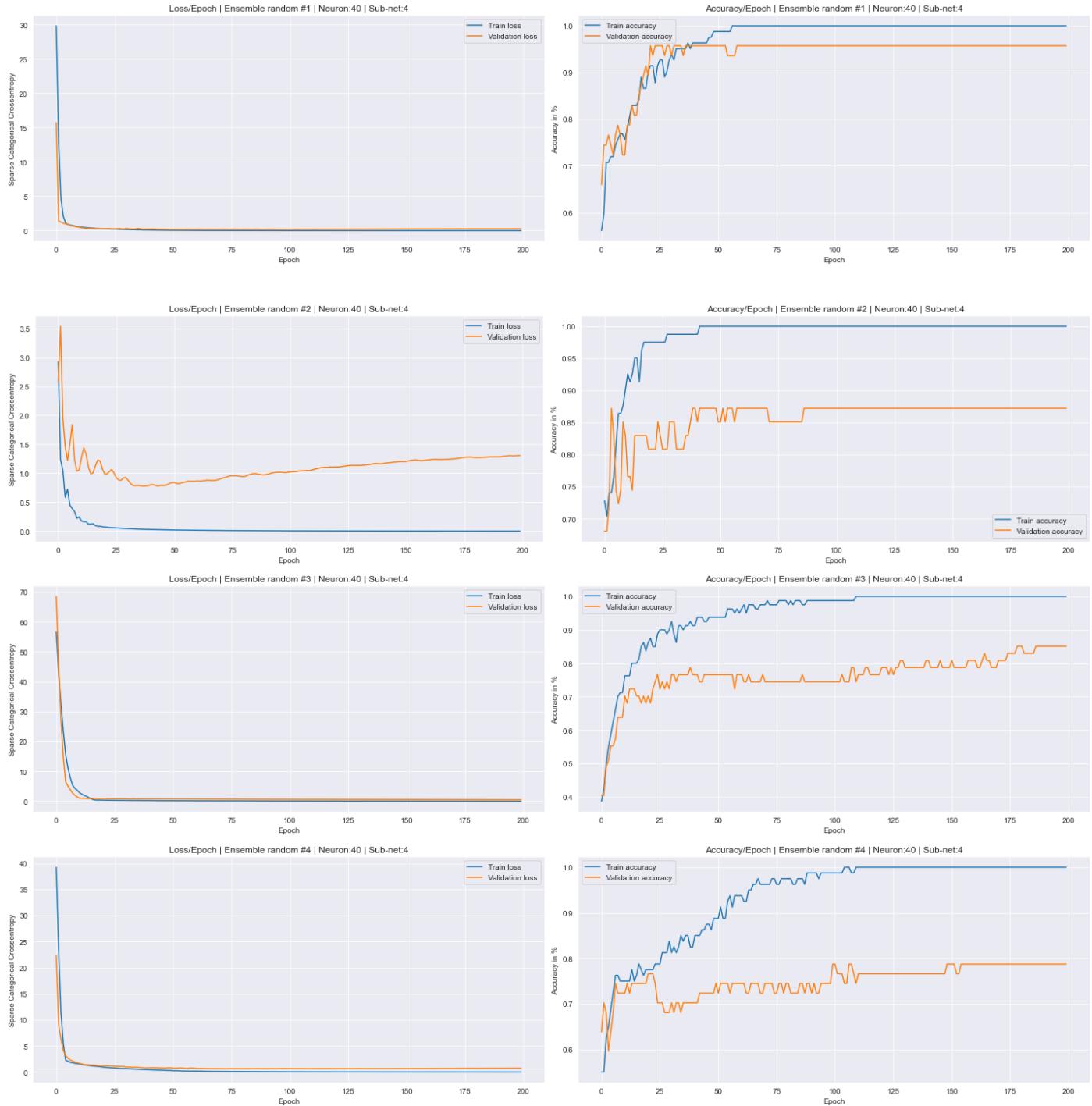
به عنوان اولین آزمایش، یک ساختار گروهی با دو زیر شبکه المن ۴۰ نورونی در نظر گرفته و داده های آموزشی را نیز به دو قسمت تقسیم کرده و هر یک از زیر شبکه ها را با آن آموزش می دهیم. نتایج حاصل از دسته بندی هایی حاصل بصورت زیر می باشد:





به عنوان دوم آزمایش، یک ساختاری گروهی با چهار زیر شبکه المن ۴۰ نورونی در نظر گرفته و داده های آموزشی را نیز به چهار قسمت تقسیم کرده و هر یک از زیر شبکه ها را با آن آموزش می دهیم. نتایج حاصل از دسته بند نهایی حاصل بصورت زیر می باشد:

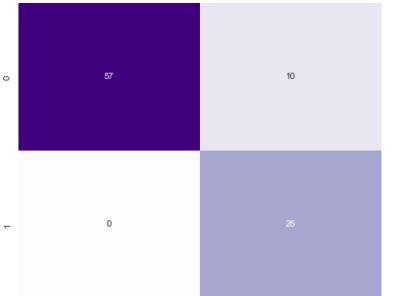


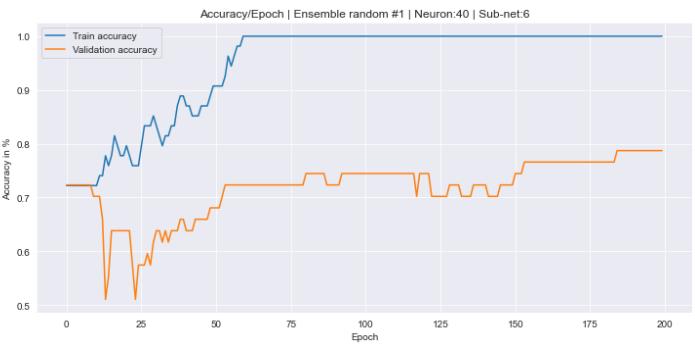
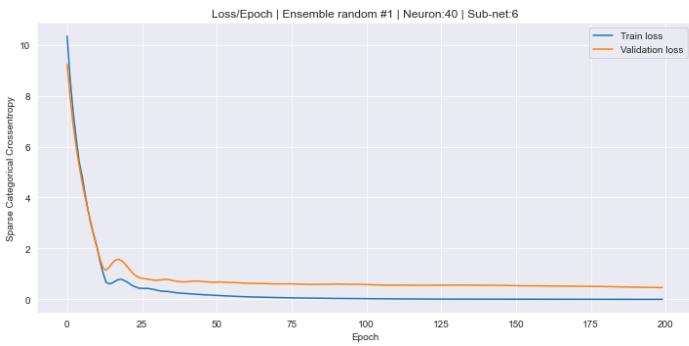
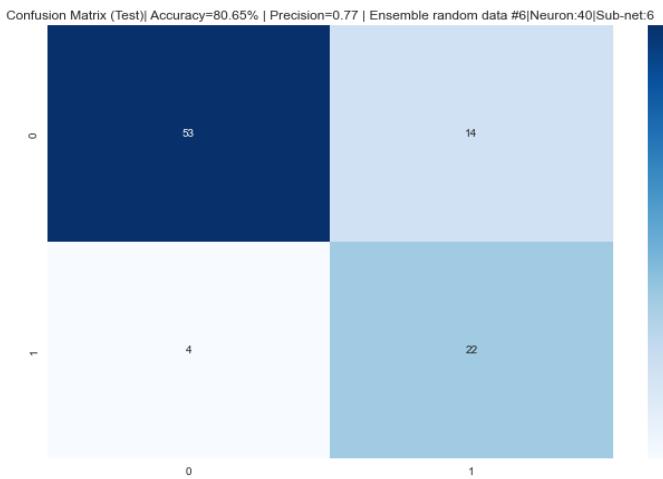
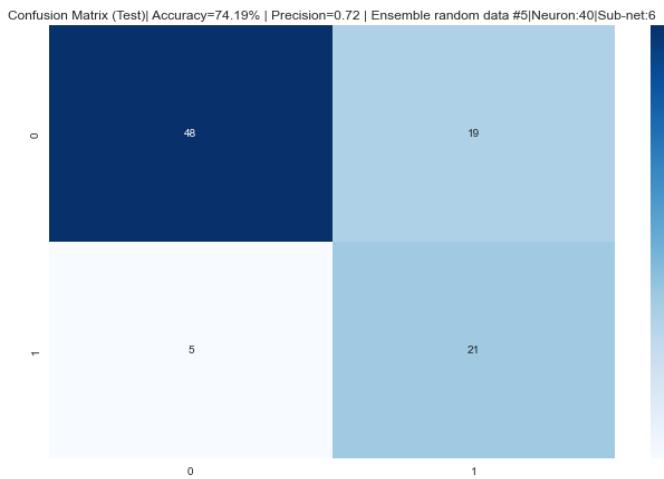
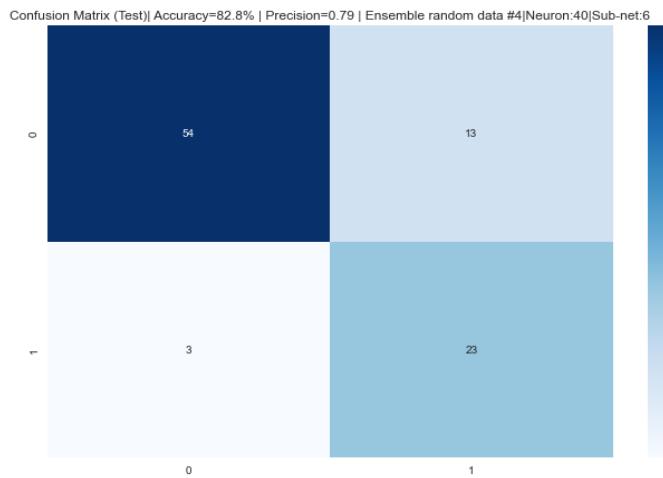
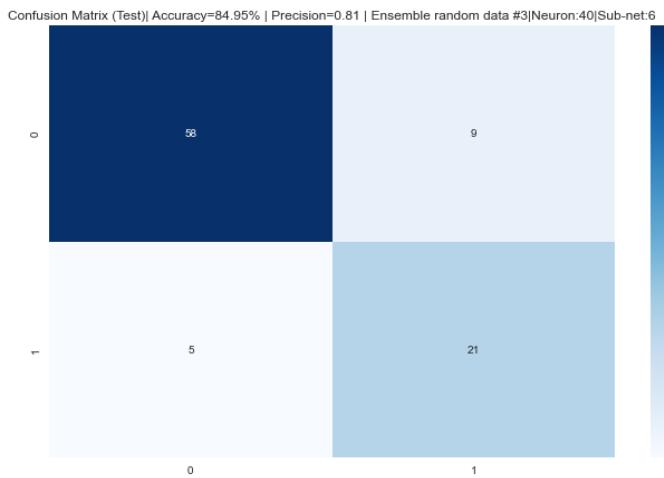
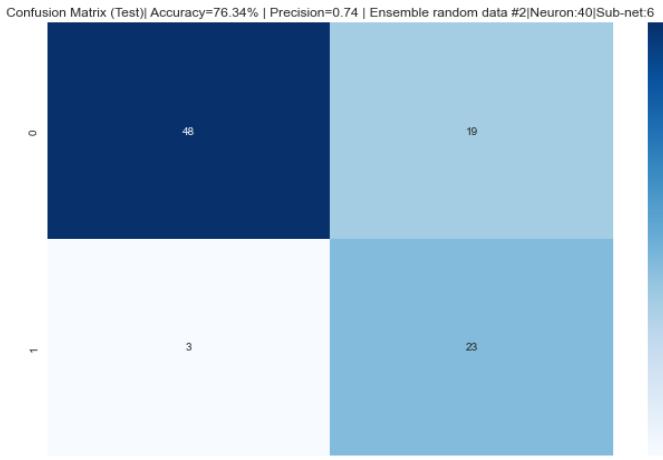
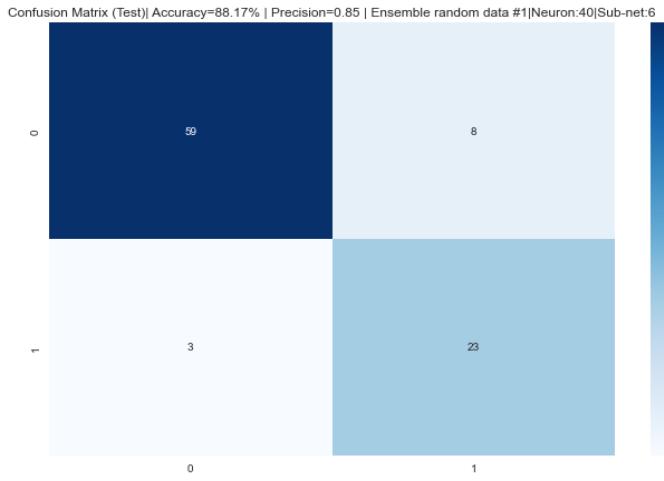


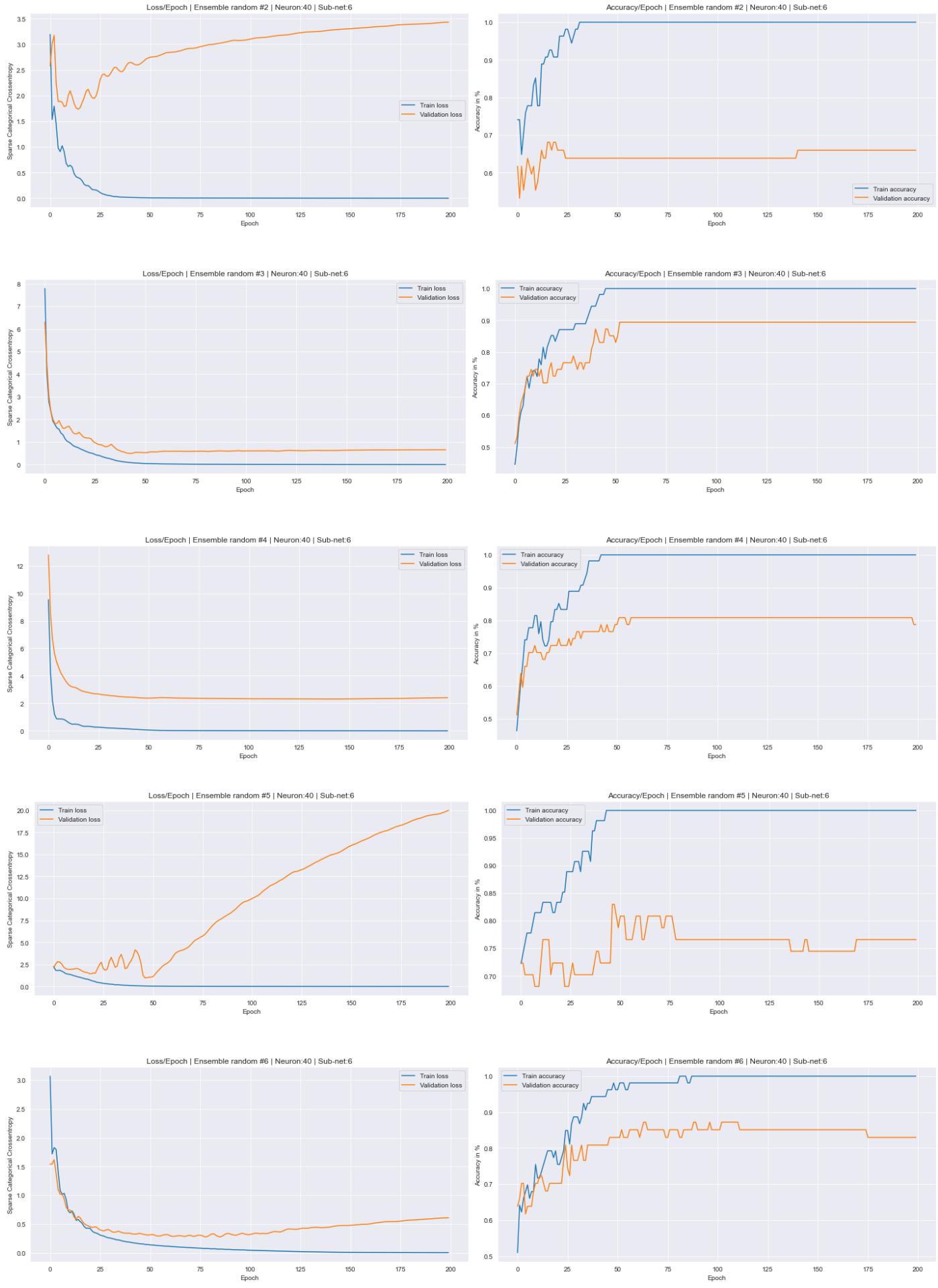
به عنوان سومین آزمایش، یک ساختاری گروهی با شش زیر شبکه المن ۴۰ نورونی در نظر گرفته و داده های آموزشی را نیز به شش قسمت

تقسیم کرده و هر یک از زیر شبکه ها را با آن آموزش می دهیم. نتایج حاصل از دسته بند نهایی حاصل بصورت زیر می باشد:

Confusion Matrix (Test) | Accuracy=89.25% | Precision=0.86 | Ensemble Elman networks with random data | Sub-net:6

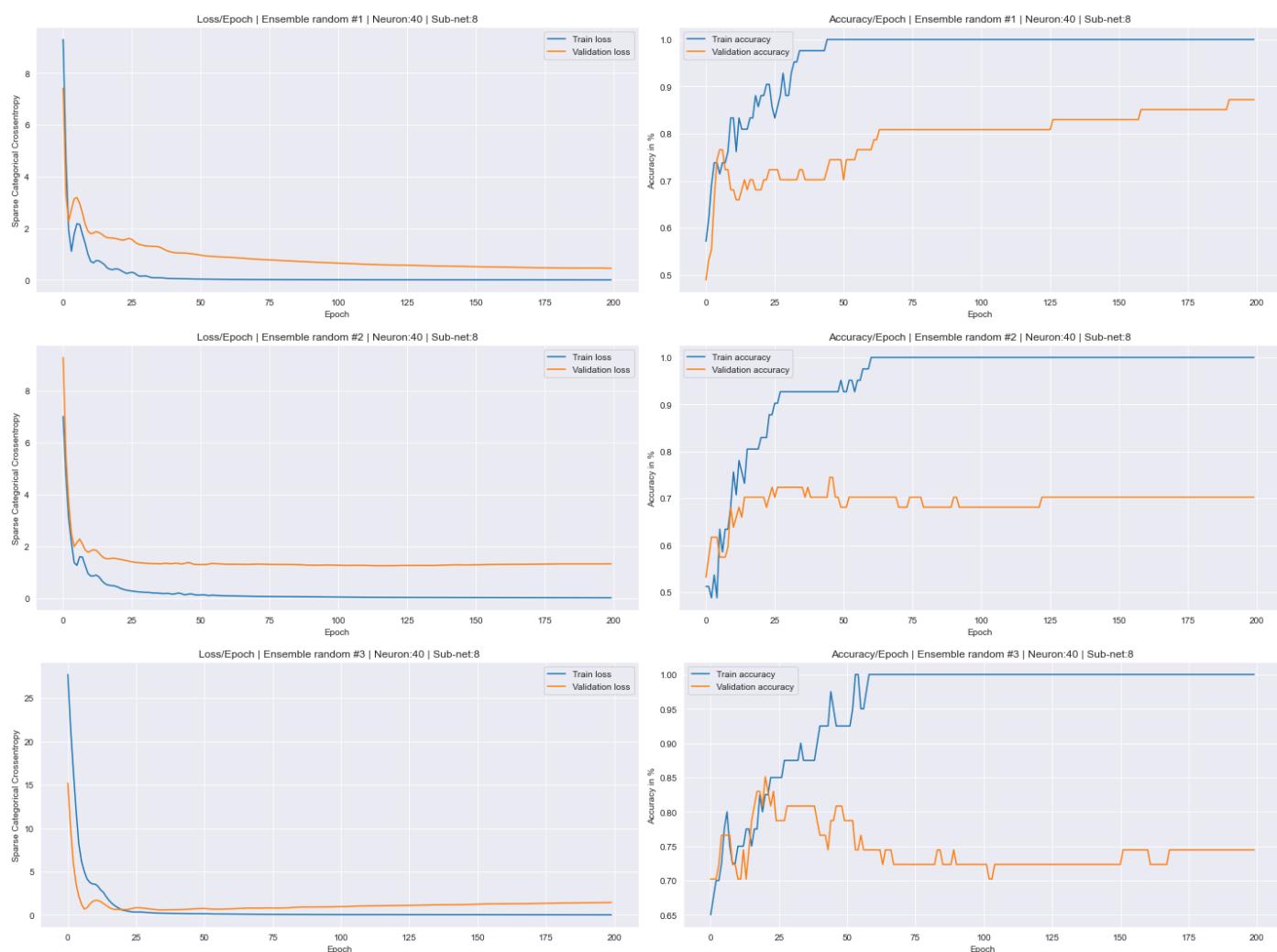
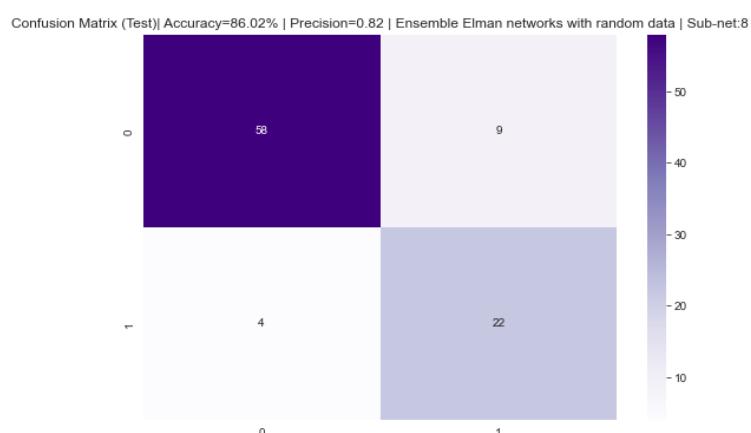


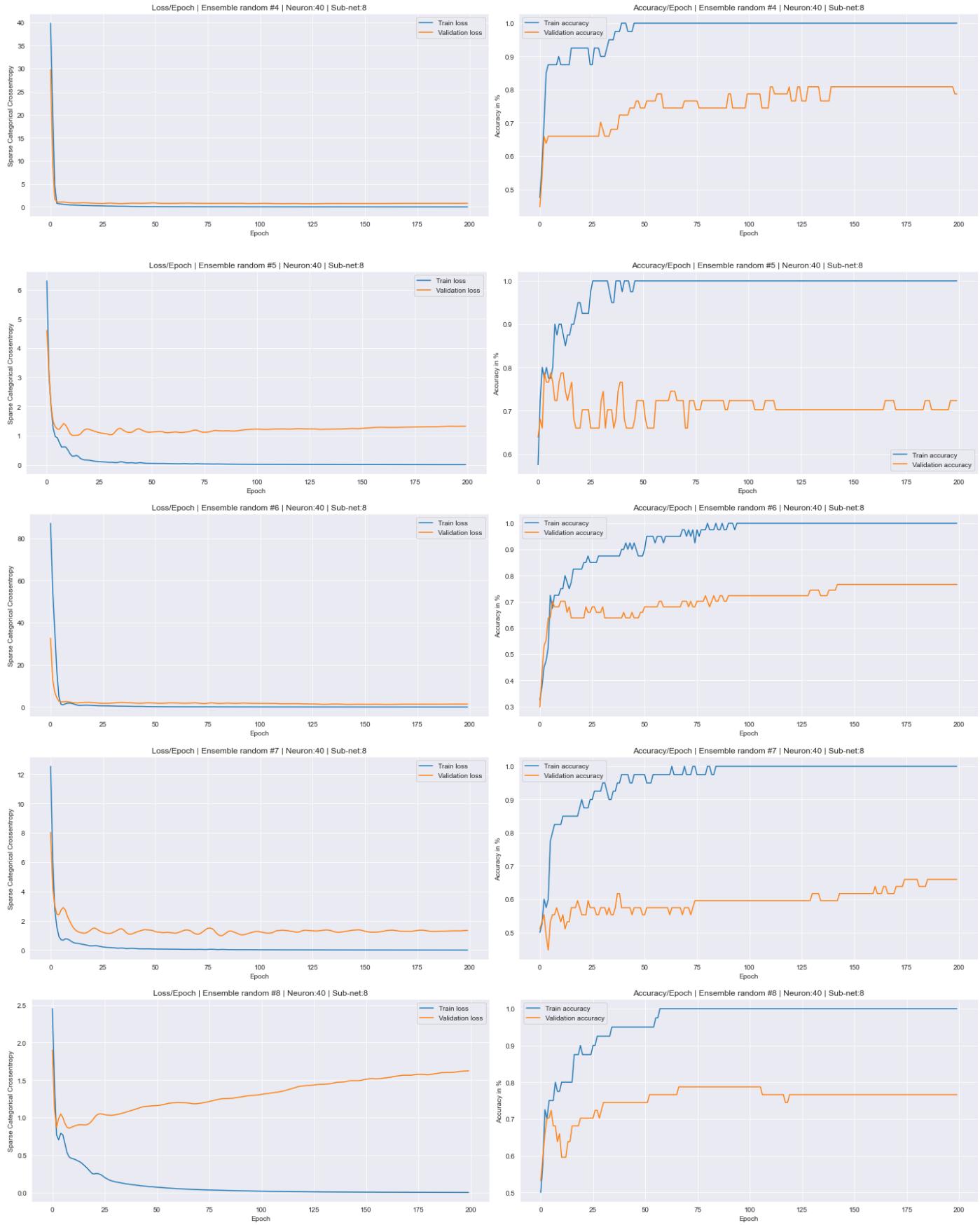




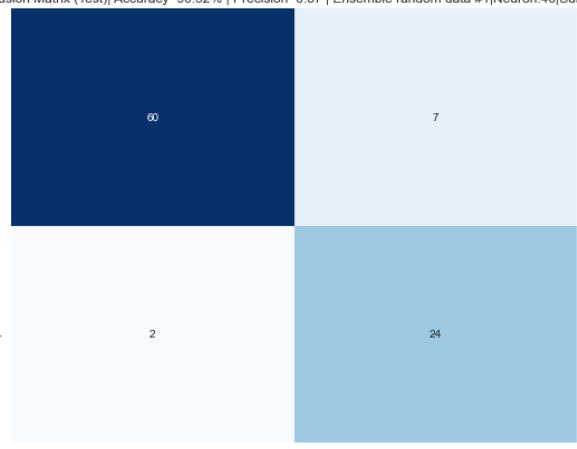
به عنوان چهارمین آزمایش، یک ساختاری گروهی با هشت زیر شبکه المن ۴۰ نورونی در نظر گرفته و داده های آموزشی را نیز به هشت

قسمت تقسیم کرده و هر یک از زیر شبکه ها را با آن آموزش می دهیم. نتایج حاصل از دسته بند نهایی حاصل بصورت زیر می باشد:

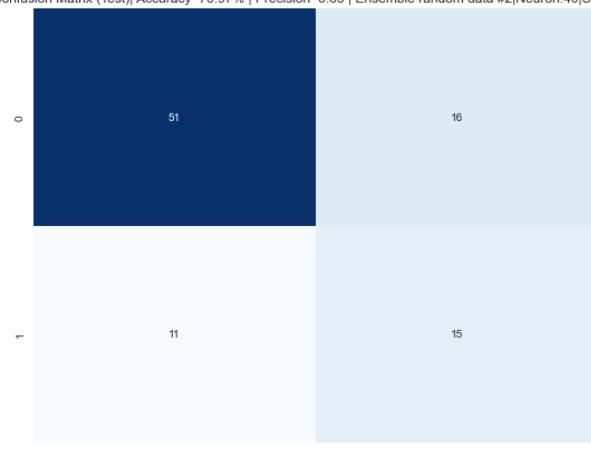




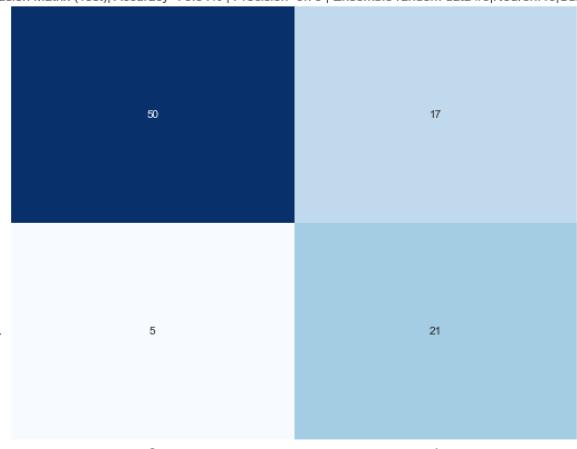
Confusion Matrix (Test) Accuracy=90.32% | Precision=0.87 | Ensemble random data #1|Neuron:40|Sub-net:8



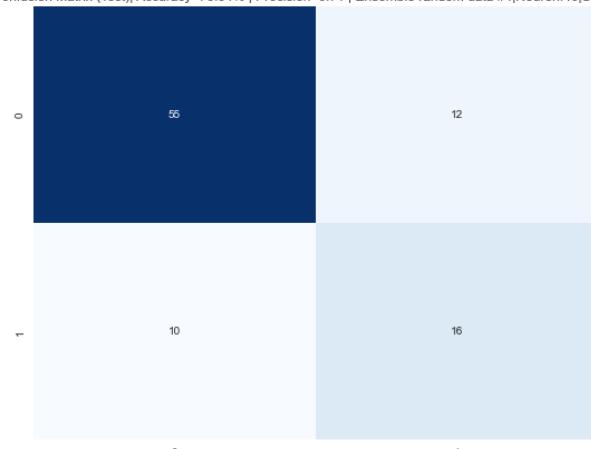
Confusion Matrix (Test) Accuracy=70.97% | Precision=0.65 | Ensemble random data #2|Neuron:40|Sub-net:8



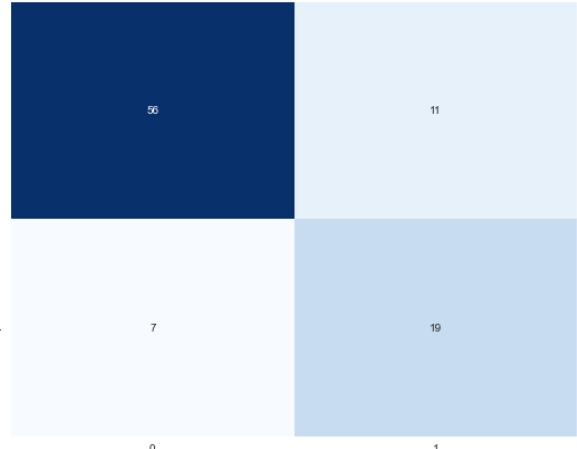
Confusion Matrix (Test) Accuracy=76.34% | Precision=0.73 | Ensemble random data #3|Neuron:40|Sub-net:8



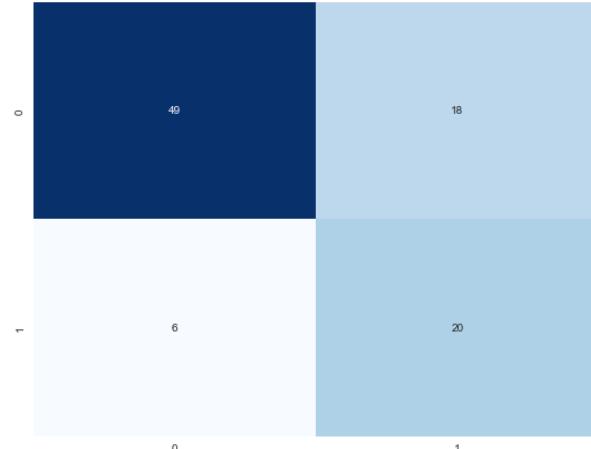
Confusion Matrix (Test) Accuracy=76.34% | Precision=0.71 | Ensemble random data #4|Neuron:40|Sub-net:8



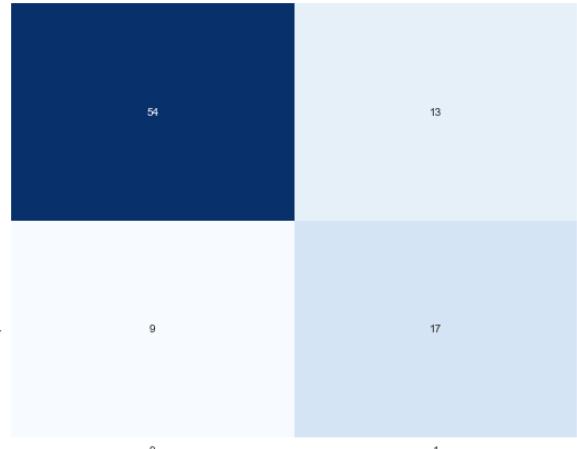
Confusion Matrix (Test) Accuracy=80.65% | Precision=0.76 | Ensemble random data #5|Neuron:40|Sub-net:8



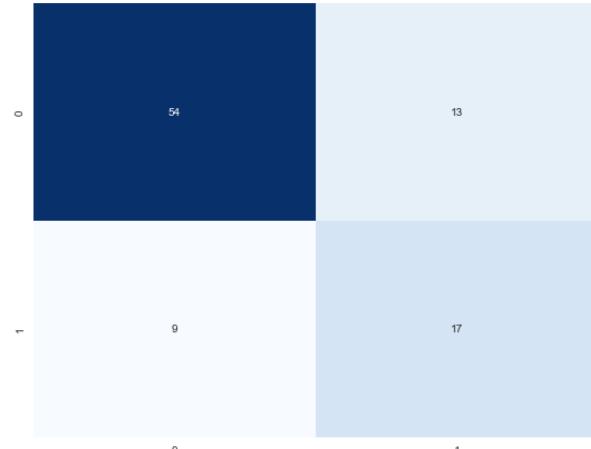
Confusion Matrix (Test) Accuracy=74.19% | Precision=0.71 | Ensemble random data #6|Neuron:40|Sub-net:8



Confusion Matrix (Test) Accuracy=76.34% | Precision=0.71 | Ensemble random data #7|Neuron:40|Sub-net:8



Confusion Matrix (Test) Accuracy=76.34% | Precision=0.71 | Ensemble random data #8|Neuron:40|Sub-net:8



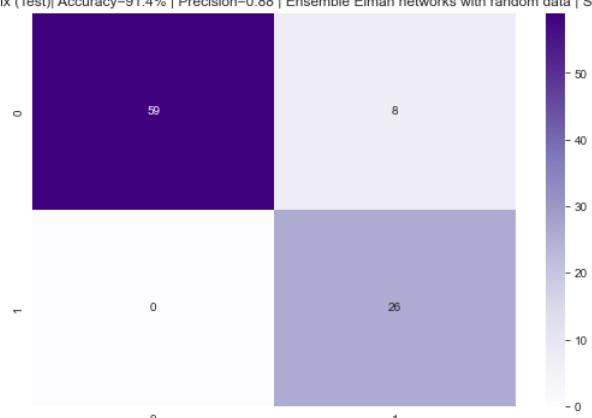
به عنوان پنجمین آزمایش، یک ساختاری گروهی با چهار زیر شبکه المن در نظر میگیریم که تفاوت آن نسبت به آزمایش بالاتر تعداد نورون

های آن می باشد که ۲۰ نورونی در نظر گرفته و داده های آموزشی را نیز به چهار قسمت تقسیم کرده و هر یک از زیر شبکه ها را با آن

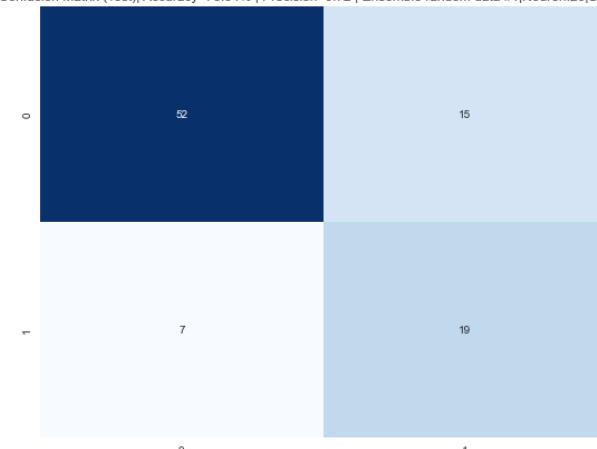
آموزش می دهیم. نتایج حاصل بصورت زیر می باشد(این آزمایش و آزمایش بعدی به جهت بررسی پیچیدگی

زیرشبکه ها در پخش تصادفی داده ها انجام شده است):

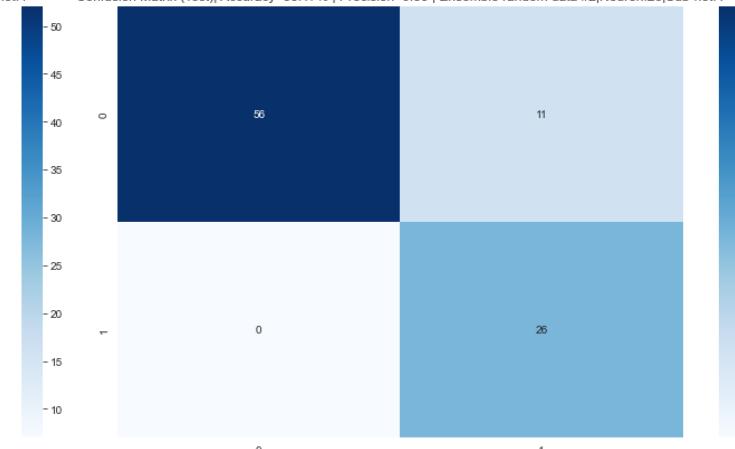
Confusion Matrix (Test) | Accuracy=91.4% | Precision=0.88 | Ensemble Elman networks with random data | Sub-net:4



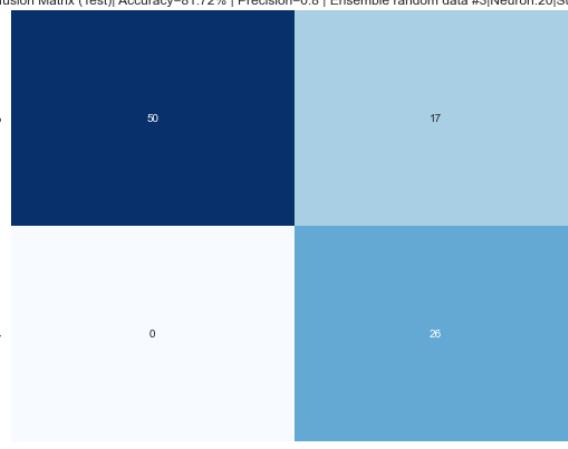
Confusion Matrix (Test) | Accuracy=76.34% | Precision=0.72 | Ensemble random data #1|Neuron:20|Sub-net:4



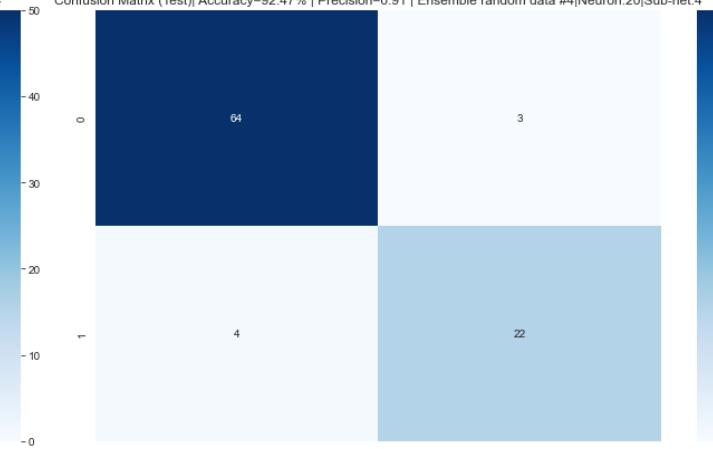
Confusion Matrix (Test) | Accuracy=88.17% | Precision=0.85 | Ensemble random data #2|Neuron:20|Sub-net:4

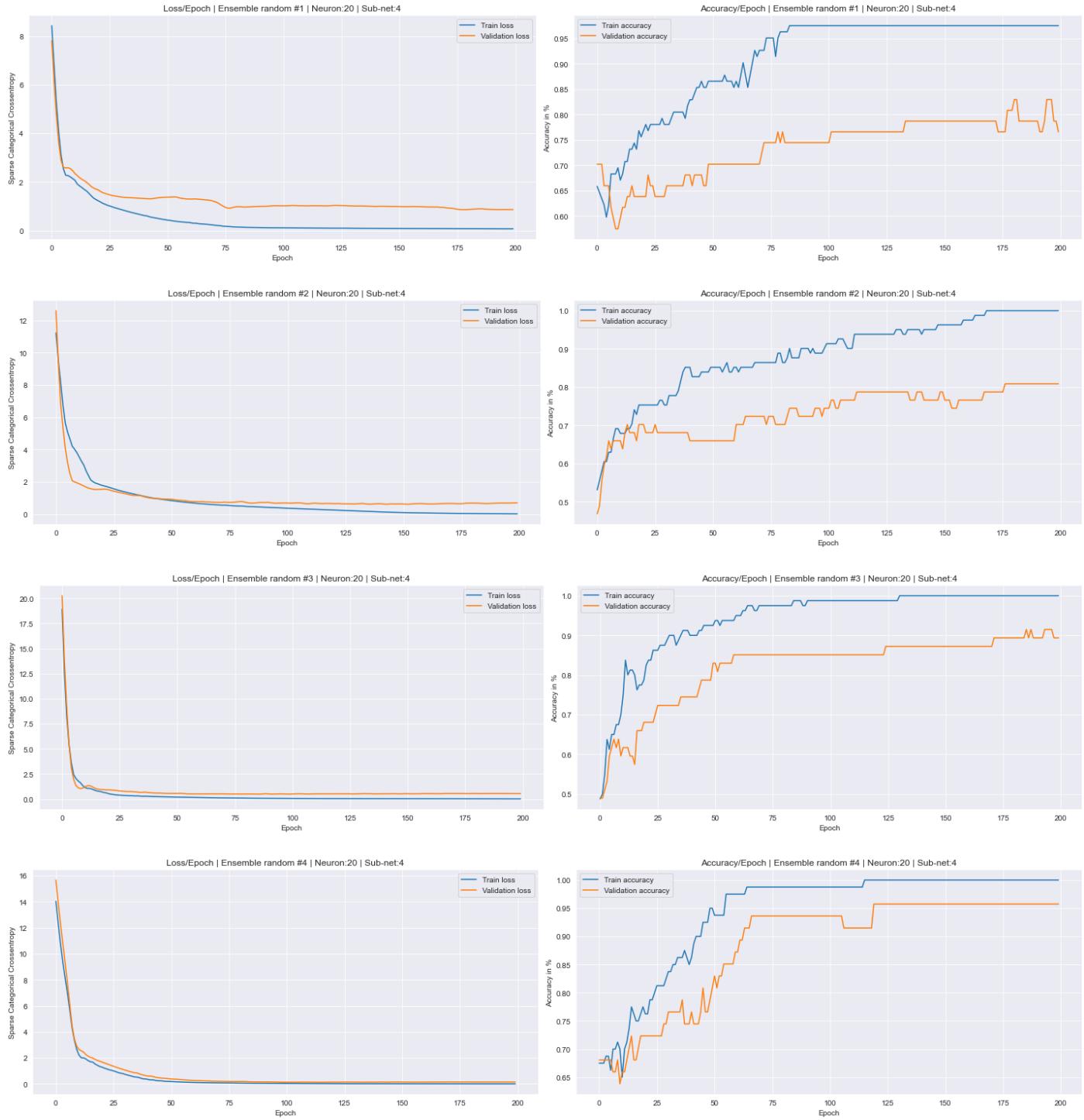


Confusion Matrix (Test) | Accuracy=81.72% | Precision=0.8 | Ensemble random data #3|Neuron:20|Sub-net:4

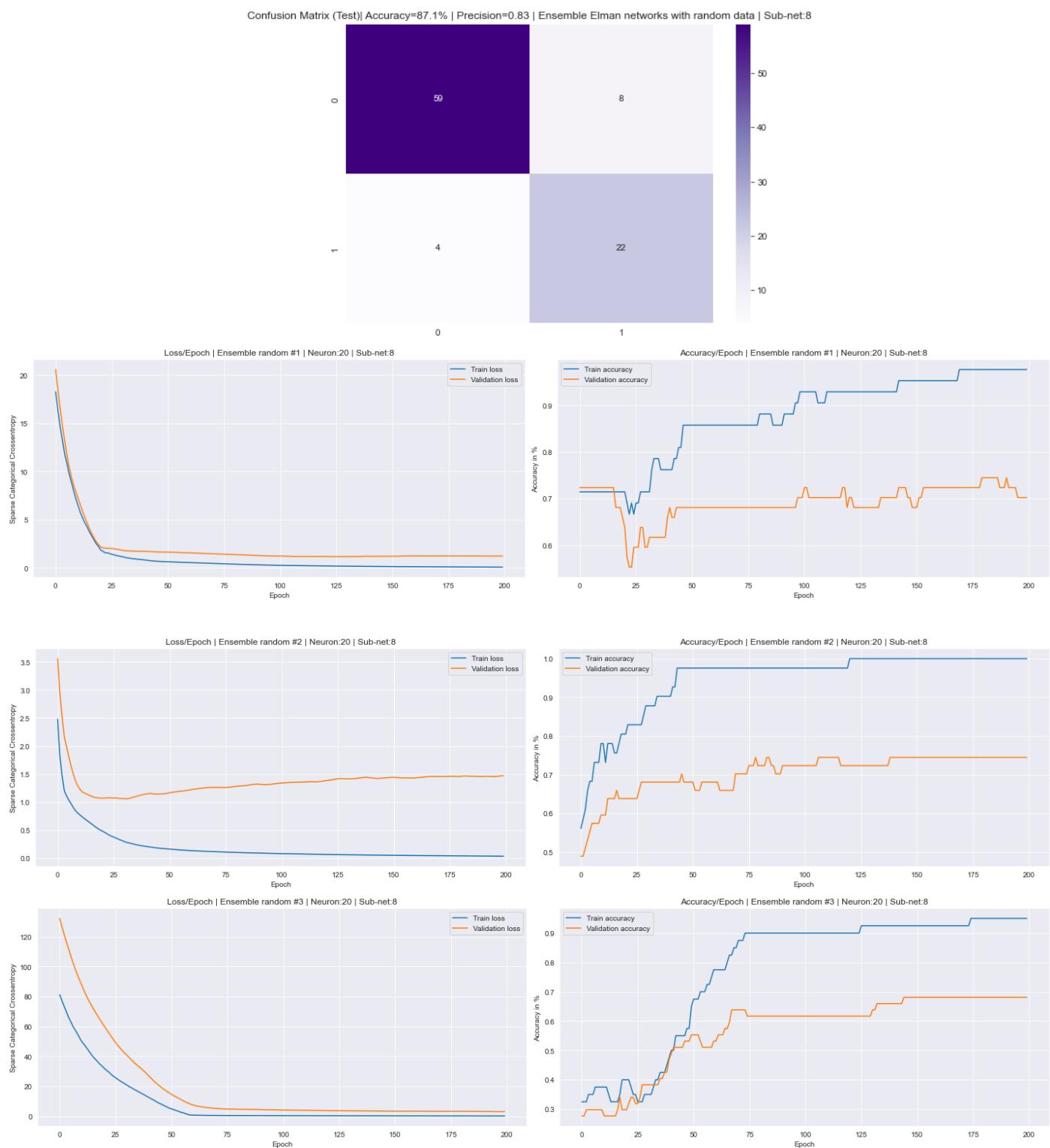


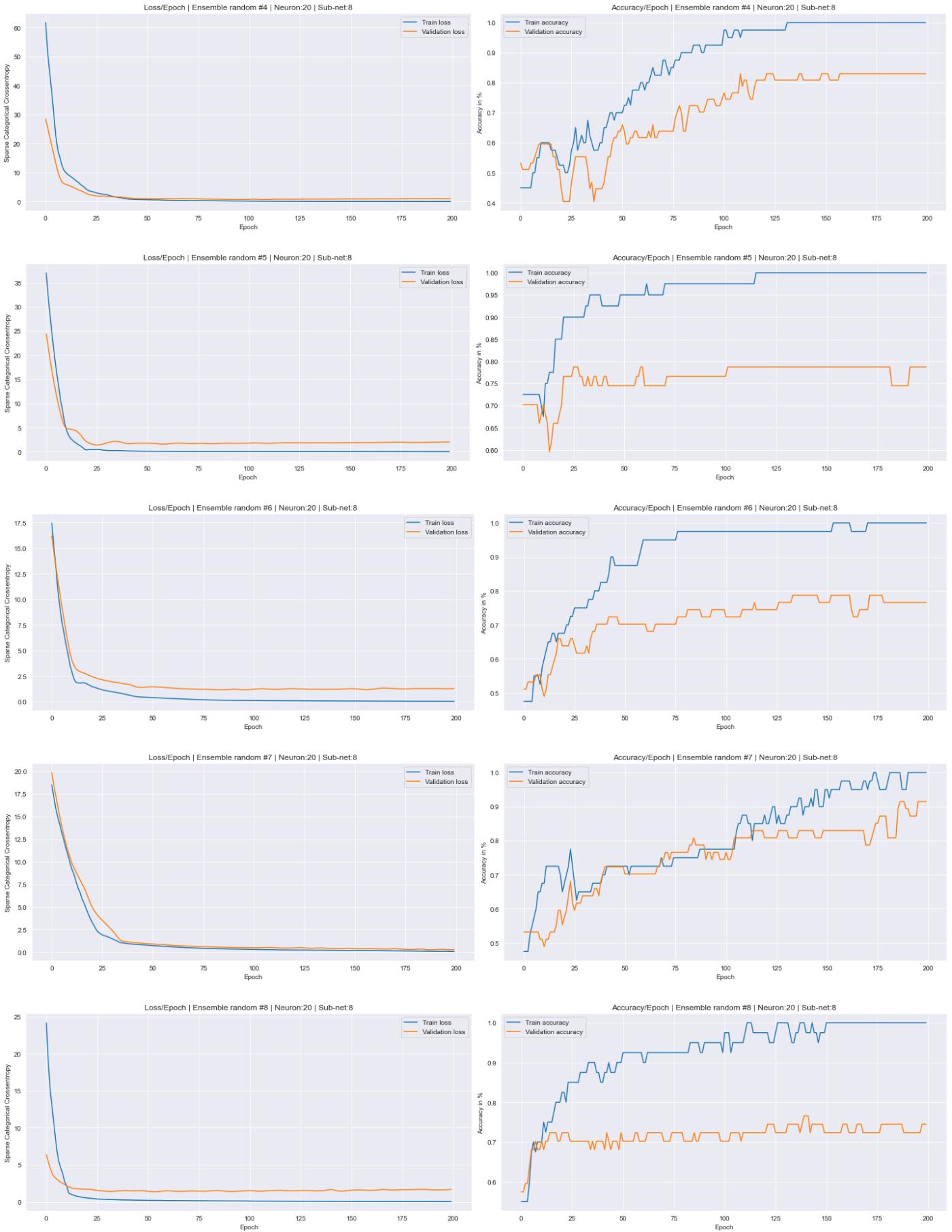
Confusion Matrix (Test) | Accuracy=92.47% | Precision=0.91 | Ensemble random data #4|Neuron:20|Sub-net:4

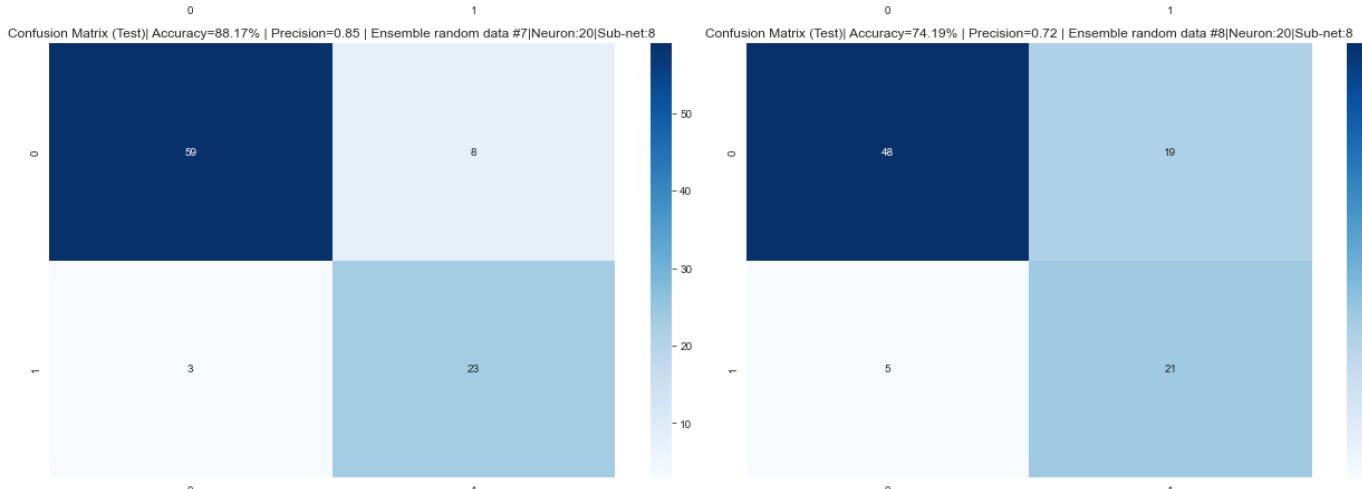
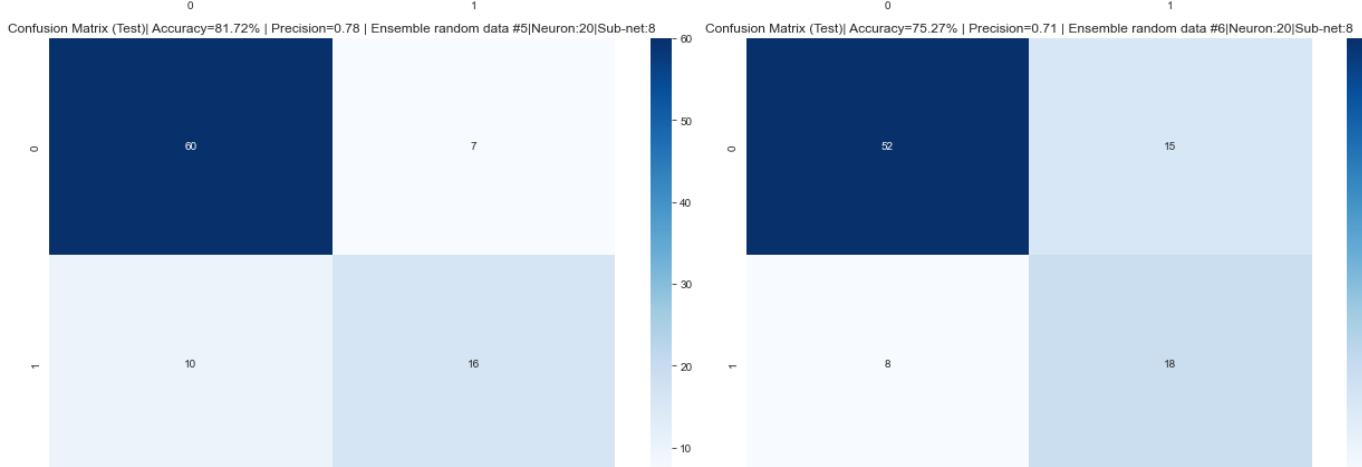
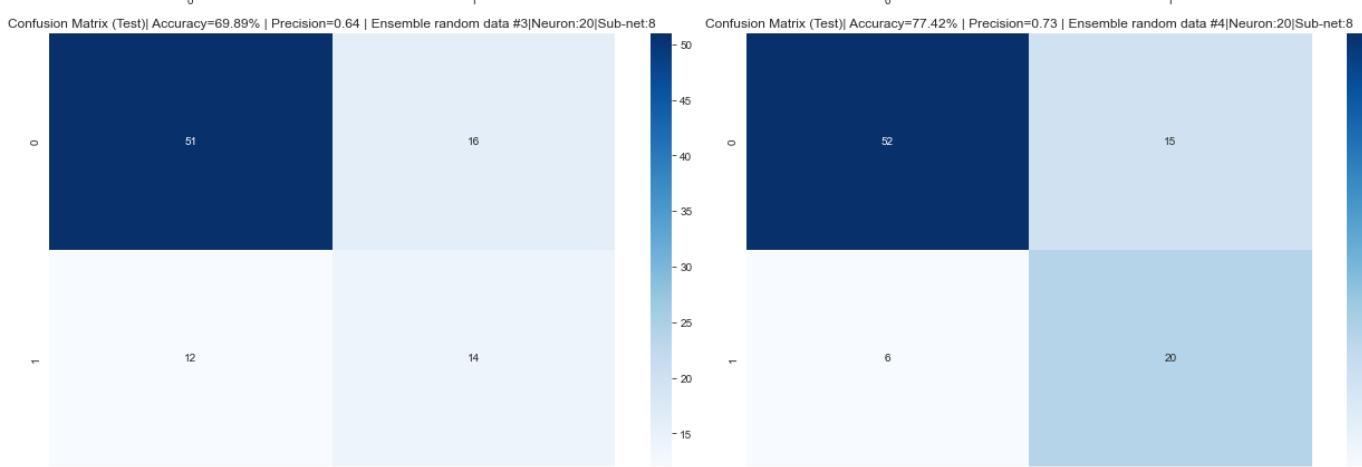
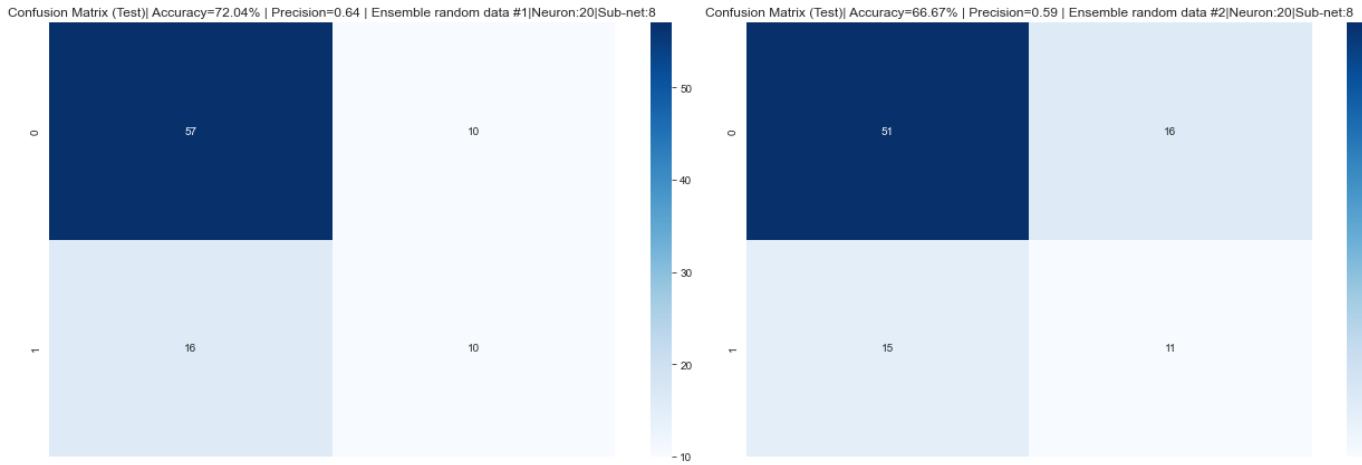




به عنوان ششمین آزمایش، یک ساختاری گروهی با هشت زیر شبکه‌من در نظر می‌گیریم که تفاوت آن نسبت به آزمایش بالاتر تعداد نورون‌های آن می‌باشد که ۲۰ نورونی در نظر گرفته و داده‌های آموزشی را نیز به هشت قسمت تقسیم کرده و هر یک از زیر شبکه‌ها را با آن آموزش می‌دهیم. نتایج حاصل از دسته‌بند نهایی حاصل بصورت زیر می‌باشد:







## جمع بندی نتایج

بر اساس تمامی نتایج آزمایشات انجام شده در تمامی بخش های این سری از تمرینات، می توان گفت شبکه المن به واسطه بازخورد بیشتر نسبت به شبکه جردن توانسته است در شرایط یکسان (به جهت تعداد نورون های لایه مخفی) نتایج بهتری را کسب کند. بهترین نتیجه کسب شده برای شبکه المن ۹۷٪ و برای شبکه جردن ۸۹٪ حاصل شده است که در هر دو مورد تعداد نورون های لایه مخفی برابر با ۵۰ بوده است.(نتایج گزارش شده سطح میانگین بوده و در تکرار های متفاوت تا ۳ درصد این عملکرد نوسان داشته اند).

در ساختار های گروهی نیز زیر شبکه های المنسنی نتیجه بهتری داشته و توانسته اند تقریبا بدون بیش برآراش هیچ کدام از زیر شبکه ها به دقت ۱۰۰٪ برای داده های unseen برسند در صورتی که این عملکرد برای شبکه های جردنی ۹۰.۳۲٪ بوده است.(البته برخی زیر شبکه ها تا ۹۲ درصد نیز عملکرد داشته اند) همچنین بر اساس نتایج خروجی شبکه های Ensemble میتوان دریافت با ترکیب چند مدل(هرچند ضعیف) میتوان به دقت های بالاتری دست یافت که در برخی از آزمایش ها نتیجه ساختار گروهی از تک تک زیرشبکه ها بهتر بوده است. در ساختار های گروهی ترکیبی نیز با انتخاب زیر شبکه های با عملکرد بهتر از هر یک از المنسنی و جردن توانستیم به دقت ۱۰۰٪ دست یابیم. در ساختار گروهی مذکور از چهار مورد از المنسنی و دو مورد از جردن به عنوان زیرشبکه استفاده شده بود.

در ساختار های گروهی که داده های آموزشی بصورت تصادفی بین زیر شبکه ها پخش شده اند، ملاحظه کردیم که با ثابت نگه داشتن عماری زیر شبکه ها، افزایش تعداد زیر شبکه ها باعث کاهش عملکرد دسته بند می شود؛ دلیل این امر آن است که داده های آموزشی تصادفی تقسیمی بین زیرشبکه ها کاهش پیدا می کند و این کاهش تعداد داده آموزشی با ثابت بودن عماری زیرشبکه ها، باعث می شود پیچیدگی زیرشبکه ها برای آن تعداد داده بسیار زیاد باشد و آموزش نتواند صحیح انجام گیرد.