



بسمه تعالی

دانشگاه صنعتی شریف

دانشکده مهندسی برق

هوش مصنوعی و محاسبات زیستی

امیرحسین صفری

گزارش پروژه درس

## فاز اول پروژه

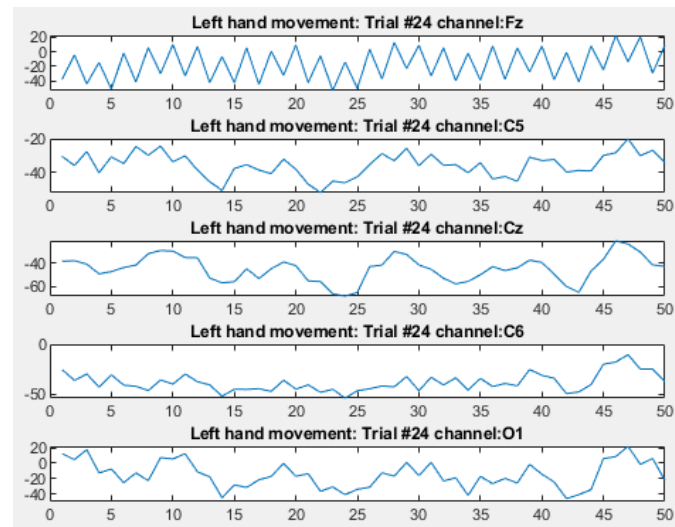
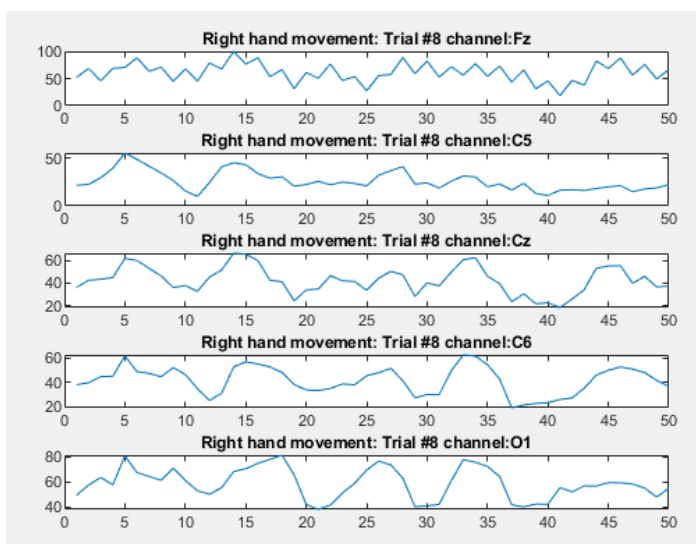
### مقدمه :

۱. در فاز ۱ پروژه ، علاوه بر داده های داده شده ، فایل های کد `CI_project_phase1.m` ، `feature_extraction.m` و `fisher_multi_dimensional.m` مورد استفاده قرار گرفته و در نهایت نیز داده های `mlp_predict_y_test.mat` و `rbf_predict_y_test.mat` که به ترتیب با استفاده از بهترین شبکه های `MLP` و `RBF` که در طول این فاز به دست آمده اند و مشخص کننده ی لیبیل داده های تست هستند ، ذخیره می شوند .

۲. در ادامه ، بخش های مختلف فاز اول پروژه و فرآیند طی شده در آن ، متناسب با ترتیبی که در کد `CI_project_phase1.m` آمده است به طور خلاصه توضیح داده شده و البته نتایج مختلف با دقت بیشتری بیان می شود .

### بخش ۱ : لود داده ها و نمایش تعدادی از کانال ها برای دو کلاس

در اولین بخش ، پس از لود داده ها ، داده های ۵ کانال `Fz` , `C5`, `Cz`, `C6`, `O1` که با استفاده از شیوه ی جایگیری کانال ها روی مغز از بخش های مختلف آن انتخاب شده ، را برای داده ی ۸ام به عنوان نماینده ی داده های با کلاس ۱ ( حرکت دست راست ) و داده ی ۲۴ام به عنوان نماینده ی داده های با کلاس صفر ( حرکت دست چپ ) نمایش داده می شود .



نمایش داده های ۵ کانال برای یکی از داده های دو کلاس

## بخش دوم: استخراج ویژگی

در این بخش با استفاده از تابع `feature_extraction.m`، ویژگی های مورد نظر استخراج شده اند.

ویژگی های آماری و فرکانسی که برای هر داده استخراج شده است، به شرح زیر می باشد:

ویژگی های آماری: ( که در مجموع شامل ۴۸۰ ویژگی می شود )

۱. واریانس ۲۸ کانال: که ویژگی های ۱ تا ۲۸ را تشکیل می دهند.
۲. چولگی (skewness) ۲۸ کانال: که ویژگی های ۲۹ تا ۵۶ را تشکیل می دهند.
۳. کشیدگی (kurtosis) ۲۸ کانال: که ویژگی های ۵۷ تا ۸۴ را تشکیل می دهند.
۴. همبستگی دو به دو کانال ها: که ویژگی های ۸۵ تا ۴۶۲ را تشکیل می دهند. ( ۳۷۸ ویژگی )
۵. میزان انرژی آنالوپی ۲۸ کانال: که ویژگی های ۴۶۳ تا ۴۹۰ را تشکیل می دهند.

ویژگی های فرکانسی: ( که در مجموع شامل ۲۹۰ ویژگی می شود )

۱. فرکانس میانگین ۲۸ کانال: که ویژگی های ۴۹۱ تا ۵۱۸ را تشکیل می دهند.
۲. فرکانس میانه ی ۲۸ کانال: که ویژگی های ۵۱۹ تا ۵۴۶ را تشکیل می دهند.
۳. Occupied Bandwidth ۲۸ کانال ( پهنای باندی که ۹۹ درصد انرژی سیگنال در آن قرار دارد): که ویژگی های ۵۴۷ تا ۵۷۴ را تشکیل می دهند.

۴. انرژی باندهای فرکانسی ۷ گانه برای هر یک از ۲۸ کانال : که ویژگی های ۵۷۵ تا ۷۷۰ را تشکیل می دهند . (۱۹۶ ویژگی)

### بخش سوم : نرمال سازی ویژگی های استخراج شده

با استفاده از دستور mapminmax ویژگی های استخراج شده را نرمال می کنیم .

### بخش چهارم : محاسبه ی مقدار J دو نوع ویژگی های آماری و فرکانسی بر اساس معیار فیشر تک بعدی و انتخاب ۴۰ تا از ویژگی ها

در این بخش برای هر دو نوع ویژگی های آماری و فرکانسی به صورت جداگانه بر اساس معیار فیشر مقدار J را به دست آورده و آن ها را بر اساس معیار به دست آمده به صورت نزولی مرتب می کنیم .

سپس با این هدف که از بین تمام ویژگی ها تنها ۴۰ ویژگی را انتخاب کنیم ؛ حالت های مختلفی را برای انتخاب بهترین ویژگی ها از هر دسته در نظر می گیریم .

حالت ها به شرح زیر می باشد :

۱. ۸ ویژگی برتر از ویژگی های فرکانسی و ۳۲ ویژگی برتر از ویژگی های آماری .
۲. ۱۶ ویژگی برتر از ویژگی های فرکانسی و ۲۴ ویژگی برتر از ویژگی های آماری .
۳. ۲۴ ویژگی برتر از ویژگی های فرکانسی و ۱۶ ویژگی برتر از ویژگی های آماری .
۴. ۳۲ ویژگی برتر از ویژگی های فرکانسی و ۸ ویژگی برتر از ویژگی های آماری .

سپس با در نظر گرفتن هر یک از حالات فوق ، از معیار فیشر در چند بعد و با استفاده از تابع fisher\_multi\_dimensional.m ، مقدار J\_score را برای تمام ۴۰ ویژگی به دست آورده و در نهایت آن حالتی که منجر به تولید بیشینه مقدار J\_score می شود را انتخاب می نماییم .

نتیجه : در نهایت ، حالت ۳ ( ۲۴ ویژگی برتر از ویژگی های فرکانسی و ۱۶ ویژگی برتر از ویژگی های آماری ) با J\_score برابر با ۰.۰۳۴۶ به عنوان بهترین ترکیب برای داشتن ۴۰ ویژگی به دست آمد و این ویژگی ها در ماتریس selected\_Train\_Features قرار می گیرند .

هم چنین بهترین ویژگی های به دست آمده دارای شماره های زیر می باشد: (با استفاده از ماتریس های  
(selected frequency features ind و selected statistical features ind

ویژگی های آماری :

ویژگی های شماره ی ۲۷ ، ۲۵۵ ، ۲۶۲ ، ۲۶۳ ، ۲۸۱ ، ۲۹۹ ، ۳۶۱ ، ۳۶۲ ، ۳۷۴ ، ۳۷۵ ، ۳۸۱ ، ۳۸۲ ، ۳۸۵ ، ۳۸۶ و ۳۹۷ .

ویژگی های فرکانسی (با شروع از ۱):

ویژگی های شماره ی ۵ ، ۴۱ ، ۱۲۶ ، ۱۳۲ ، ۱۳۳ ، ۱۳۴ ، ۱۳۵ ، ۱۳۹ ، ۱۴۳ ، ۱۴۴ ، ۱۵۰ ، ۱۵۱ ، ۱۵۴ ، ۱۵۷ ، ۱۵۸ ، ۱۵۹ ، ۱۶۰ ، ۱۶۱ ، ۱۶۵ ، ۱۶۶ ، ۱۸۲ ، ۱۸۶ ، ۱۸۹ ، ۱۹۳ .

### **بخش پنجم : طراحی شبکه ی MLP بر روی داده های آموزشی و یافتن بهترین آن**

در این بخش ، شبکه های MLP را با تغییر متغیر های مختلف طراحی کرده و با اعمال آن ها بر روی داده های آموزشی و استفاده از شیوه ی ارزیابی 5-fold cross-validation ، بهترین آن ها را انتخاب می کنیم .

ویژگی های این شبکه ها به شرح زیر می باشد :

۱. برای این شبکه ها ، تنها یک لایه ی پنهان در نظر می گیریم ، این انتخاب به این دلیل هست که شبکه های با دو و سه لایه هم تست شده اند ، اما با مشاهده ی مقادیر صحت طبقه بندی ، مقدار accuracy شبکه های با دو یا سه لایه کمتر از مقدار accuracy شبکه های با یک لایه هست و هم چنین چون با توجه به در نظر گرفتن توابع فعالسازی مختلف و تعداد نورورن های متفاوت ، در نظر گرفتن لایه های متغیر نیز که با توجه به تست ها تاثیر محسوسی در accuracy ندارد منجر به طولانی شدن بیش از حد زمان اجرا می شود ؛ در نتیجه تنها از یک لایه ی پنهان در کد استفاده شده است.

۲. تعداد نورون های لایه میانی را از ۱ تا ۱۰ تا به صورت متغیر در نظر می گیریم و با توجه به حلقه تو در تو بهترین این تعداد را به دست می آوریم.
  ۳. از تابع patternnet برای طراحی شبکه ها استفاده می کنیم .
  ۴. از ۶ تابع فعالسازی به شرح زیر به عنوان تابع فعالسازی نورون های لایه ی میانی استفاده می کنیم و با توجه به حلقه تو در تو بهترین این تابع را به دست می آوریم.  

$$\text{Activation functions} = \{ \text{"radbas"}, \text{"logsig"}, \text{"purelin"}, \text{"satlin"}, \text{"tansig"}, \text{"hardlims"} \}$$
  ۵. در تنظیمات اولیه شبکه ، تابع یادگیری (trainFcn) را بر روی 'trainbr' و به تبع این انتخاب ، تابع performFcn را بر روی 'mse' تنظیم می کنیم ، این تابع یادگیری نسبت به سایر توابع یادگیری مقدار صحت طبقه بندی بیشتری بر روی داده های اعتبارسنجی به دست می داد .
  ۶. برای یافتن بهترین threshhold ، از نمودار roc استفاده می شود ، این ترشهولد نسبت به  $\text{threshhold} = 0.5$  به نتایج بهتری در ختم می شود .
  ۷. با توجه به گفته های فوق در انتها یک ماتریس با بعد  $10 \times 6$  که در بردارنده ی مقدار صحت -fold cross-validation در حالت های مختلف تعداد نورون های لایه میانی و هم چنین توابع مختلف فعالسازی می باشد ، به دست آورده و بهترین حالت را به دست می آوریم.
- توضیح : این بهترین مقادیر صحت و ویژگی های به دست آمده با توجه به مقادیر اولیه ی رندمی که الگوریتم در هر بار در نظر می گیرد ، می تواند در هر بار اجرا اندکی متفاوت با مقادیری باشد که در ادامه گزارش می شود ، این گزارش مبتنی بر آخرین اجرای این کد می باشد .
- نتیجه : بهترین شبکه ی MLP به دست آمده دارای ویژگی های زیر می باشد.

۱. بهترین صحت طبقه بندی به دست آمده ( mlp\_best\_acc ) = ۶۷.۰۹ درصد
۲. بهترین تعداد نورون ها در لایه ی میانی ( mlp\_best\_neuron\_size ) = ۴
۳. بهترین تابع فعالسازی ( mlp\_best\_active\_func ) = 'tansig'

## بخش ششم: طراحی شبکه ی RBF بر روی داده های آموزشی و یافتن بهترین آن

در این بخش، شبکه های RBF را با تغییر متغیرهای مختلف طراحی کرده و با اعمال آن ها بر روی داده های آموزشی و استفاده از شیوه ی ارزیابی ۵-fold cross-validation، بهترین آن ها را انتخاب می کنیم.

ویژگی های این شبکه ها به شرح زیر می باشد:

۱. شعاع نورون های لایه ی میانی را مقادیر با شروع از ۱ و ختم به ۵ با فاصله ی ۰.۳ به صورت

متغیر در نظر می گیریم و با توجه به حلقه تو در تو بهترین این تعداد را به دست می آوریم.

۲. حداکثر تعداد نورون های لایه میانی را در بین مقادیر زیر به صورت متغیر در نظر می گیریم و با

توجه به حلقه تو در تو بهترین این تعداد را به دست می آوریم.

$$\text{NMat} = [2, 5, 7, 10, 12, 15, 17, 20, 22, 25, 27]$$

۳. از تابع newrb برای طراحی این شبکه ها استفاده می کنیم.

۴. Treshhold را برابر با ۰.۵ در نظر می گیریم، بر خلاف شبکه ی MLP، ترشهود برابر با ۰.۵.

نسبت به ترشهودی که از طریق roc به دست می آید، نتایج بهتری دارد.

۵. با توجه به گفته های فوق در انتها یک ماتریس با بعد ۱۱\*۱۴ که در بردارنده ی مقدار صحت 5-

fold cross-validation در حالت های مختلف تعداد نورون های لایه میانی و هم چنین توابع

مختلف فاعاسازی می باشد، به دست آورده و بهترین حالت را به دست می آوریم.

توضیح: این بهترین مقادیر صحت و ویژگی های به دست آمده با توجه به مقادیر اولیه ی رندمی که

الگوریتم در هر بار در نظر می گیرد، می تواند در هر بار اجرا اندکی متفاوت با مقادیری باشد که در ادامه

گزارش می شود، این گزارش مبتنی بر آخرین اجرای این کد می باشد:

نتیجه: بهترین شبکه ی RBF به دست آمده دارای ویژگی های زیر می باشد.

۱. بهترین صحت طبقه بندی به دست آمده (rbf\_best\_acc) = ۶۷.۴۱ درصد

۲. بهترین مقدار برای شعاع نورون های میانی (rbf\_best\_spread) = ۴.۹

۳. بهترین مقدار برای حداکثر تعداد نورون های میانی (rbf\_best\_N) = ۱۰

### مقایسه ی بهترین شبکه ی MLP با بهترین شبکه RBF :

۱. هر دو شبکه صحتی که برای طبقه بندی داده های اعتبارسنجی ارایه می دهند تقریباً با هم برابر و در حدود ۶۷ درصد می باشد .

۲. هر دو شبکه ای که به دست آمده اند ، می توان گفت شبکه های پیچیده ای نیستند که منجر به overfit شدن مدل بر روی داده ها شده باشند ؛ شبکه ی MLP که تنها با یک لایه نورون پنهان و تعداد ۴ نورون در لایه ی پنهان به دست آمده و شبکه ی RBF هم با استفاده از حداکثر ۱۰ نورون به دست آمده است .

### **بخش هفتم : استخراج تمام ویژگی ها از داده های تست و سپس انتخاب ویژگی های منتخب از آن ها**

در این بخش با استفاده از تابع feature\_extraction.m ، تمام ۷۷۰ ویژگی ای که در ابتدا در نظر گرفته شد از داده ی تست استخراج می کنیم و پس از نرمال کردن این ویژگی ها با استفاده از تنظیمات xPS که در هنگام نرمال سازی داده های آموزشی به دست آوردیم ، از بین این ۷۷۰ ویژگی ، آن ۴۰ ویژگی ای که در بخش چهارم به دست آوردیم ، انتخاب می کنیم ؛ این ویژگی های منتخب در ماتریس selected\_Test\_Features قرار می گیرند.

به این ترتیب داده های تست برای اعمال بهترین شبکه های به دست آمده بر روی آن ها آماده می شوند.

### **بخش هشتم : طبقه بندی داده های تست با استفاده از بهترین شبکه ی MLP به دست آمده**

در این بخش نیز با استفاده از بهترین پارامترهایی که برای شبکه ی MLP به دست آمده است و کلیات شبکه ی در نظر گرفته شده ( مثل شیوه ی به دست آوردن ترشهولد و ... ) ، داده های تست را طبقه بندی کرده و لیبل آن ها را به دست می آوریم ، این لیبل ها در ماتریس mlp\_predict\_y\_test.mat ذخیره شده و هم چنین در کنار فایل های آپلود شده قرار دارند.

**بخش نهم: طبقه بندی داده های تست با استفاده از بهترین شبکه ی RBF به دست آمده**

در این قسمت نیز با استفاده از بهترین پارامترهایی که برای شبکه ی RBF به دست آمده است و کلیات شبکه ی در نظر گرفته شده ( مثل شیوه ی به دست آوردن ترشهولد و ... ) ، داده های تست را طبقه بندی کرده و لیبل آن ها را به دست می آوریم ، این لیبل ها در ماتریس `rbf_predict_y_test.mat` ذخیره شده و هم چنین در کنار فایل های آپلود شده قرار دارند.

**بخش دهم: ذخیره ی ویژگی های منتخب برای داده های آموزشی و تست برای استفاده در فاز ۲**

در آخرین بخش این فاز ، ماتریس های `selected_Train_Features.mat` و `selected_Test_Features.mat` را ذخیره می کنیم تا برای فاز ۲ بتوانیم از آن ها استفاده کنیم.



## فاز دوم پروژه

### مقدمه :

۱. در فاز ۲ پروژه ، علاوه بر داده های داده شده ، و ماتریس های selected\_Train\_Features.mat و selected\_Test\_Features.mat فایل های کد fisher\_multi\_dimensional.m و fisher\_fitness.m ، Cl\_project\_phase2.m مورد استفاده قرار گرفته و در نهایت نیز داده های genetic\_mlp\_predict\_y\_test.mat و genetic\_rbf\_predict\_y\_test.mat که به ترتیب با استفاده از بهترین شبکه های MLP و RBF در طول این فاز به دست آمده اند و مشخص کننده ی لیبیل داده های تست را در این فاز هستند ، ذخیره می شوند.
۲. در ادامه ، بخش های مختلف فاز دوم پروژه و فرآیند طی شده در آن ، متناسب با ترتیبی که در کد Cl\_project\_phase2.m آمده است به طور خلاصه توضیح داده شده و البته نتایج مختلف با دقت بیشتری بیان می شود .

### بخش اول : لود داده های مورد استفاده در این فاز

داده های All\_data.mat با هدف استفاده از y\_train و هم چنین ویژگی های منتخب برای داده های آموزش و تست لود می شوند.

### بخش دوم : انتخاب تعدادی ویژگی ( با تعداد بین ۱۵ الی ۳۰ ) از بین ویژگی های منتخب به دست آمده در فاز ۱ با استفاده از الگوریتم ژنتیک

در این بخش قصد داریم با استفاده از الگوریتم های ژنتیک و اسکریپت toolbox ، optimtool بتوانیم از بین ۴۰ ویژگی انتخاب شده در فاز ۱ ، تعداد کمتری ویژگی انتخاب کنیم .

ویژگی های الگوریتم مورد استفاده به صورت زیر می باشد:

۱. کروموزوم را به صورت یک درایه ی ۴۰ خانه ای در نظر می گیریم ، اگر که خانه ی i ام کروموزوم ، ۱ باشد از آن ویژگی استفاده می کنیم و در غیر اینصورت از آن ویژگی استفاده نمی کنیم .
۲. از تابع `fisher_fitness` به عنوان تابع ارزیابی استفاده می کنیم ( این تابع در ادامه توضیح داده می شود )
۳. نوع `PopulationType` را بر روی `bitstring` تنظیم می کنیم .
۴. از حالت انتخاب را بر روی `roulette` قرار داده .
۵. از جهش یکنواخت به عنوان تابع جهش و از `crossover` دو نقطه ای استفاده می کنیم .
۶. معیار خاتمه ی الگوریتم را نیز حالتی قرار می دهیم که اگر مقدار تفاوت تابع ارزیابی کمتر از یک هزارم شد ، الگوریتم متوقف شود .

توضیح تابع ارزیابی `fisher_fitness` :

در این تابع نیز از معیار `fisher` در حالت چندبعدی استفاده می کنیم ، به این صورت که پس از دریافت کروموزوم `X` و در نظر گرفتن درایه هایی که مقدار ۱ دارند ، ویژگی های متناظر با این کروموزوم `X` را به دست می آوریم و مقدار `J_score` آن ویژگی ها را با استفاده از تابع `fisher_multi_dimensional.m` به دست می آوریم .

هم چنین یک قید برای کروموزم ها در نظر می گیریم که تعداد ویژگی های انتخاب شده حتما بین ۱۵ تا ۳۰ ویژگی باشد و در صورتی که در این محدوده نباشد یک مقدار خطا به مقدار به دست آمده با `J_score` اضافه می کنیم و در نهایت مقدار `score` را خروجی می دهیم ( توضیح : با توجه به اینکه ابزار `optimtool` سعی دارد که مقدار خروجی تابع ارزیابی را کمینه کند ما مقدار `J_score` را قرینه می کنیم )

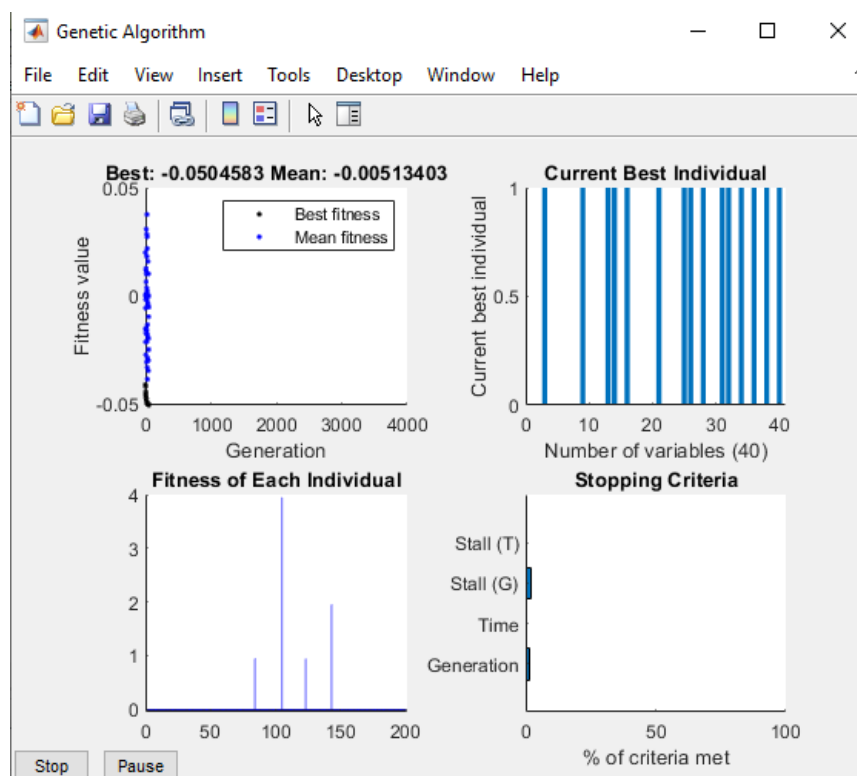
پس از اعمال تنظیمات فوق و اجرای تابع `ga` ، در نهایت یک آرایه ی نهایی با نام `genetic_ind` خروجی می گیریم که بهترین کروموزومی است که الگوریتم با توجه به قید ها به آن رسیده است و با توجه به این کروموزوم ویژگی هایی که توسط الگوریتم ژنتیک به دست آمده اند از آن ۴۰ ویژگی جدا

کرده و آن‌ها را برای داده‌های آموزش و تست به دست می‌آوریم، که به ترتیب در دو ماتریس `genetic_Test_Features` و `genetic_Train_Features` نگهداری می‌کنیم.

نتیجه: از ۴۰ ویژگی، ۱۵ ویژگی را در نهایت انتخاب می‌کنیم، این ۱۵ ویژگی، ویژگی‌های با اندیس ۳، ۹، ۱۳، ۱۴، ۱۶، ۲۱، ۲۵، ۲۶، ۲۸، ۳۱، ۳۲، ۳۴، ۳۶، ۳۸ و ۴۰ از بین ۴۰ ویژگی انتخاب شده می‌باشد که در صورتی که بخواهیم بگوییم این ویژگی‌ها متناظر با کدام ویژگی‌های آماری و فرکانسی اولیه است داریم:

ویژگی‌های آماری: ویژگی‌های شماره ۲۶۲، ۳۷۴، ۳۸۵، ۳۸۶ و ۳۹۷.

ویژگی‌های فرکانسی (با شروع از ۱): ویژگی‌های شماره ۱۳۳، ۱۴۳، ۱۴۴، ۱۵۱، ۱۵۸، ۱۵۹، ۱۶۱، ۱۶۶، ۱۸۶، ۱۹۳



نمایی از صفحه‌ی ابزار Genetic Algorithm

### بخش سوم: طراحی شبکه ی MLP بر روی داده های آموزشی و یافتن بهترین آن

در این بخش دقیقاً مشابه و مطابق با جزییات و ویژگی هایی که در بخش ۵ گزارش فاز ۱ بیان شد، عمل می کنیم با این تفاوت که در این بخش از ویژگی های genetic\_Train\_Features استفاده کرده و بهترین پارامترها را به دست می آوریم.

توضیح: این بهترین مقادیر صحت و ویژگی های به دست آمده با توجه به مقادیر اولیه ی رندمی که الگوریتم در هر بار در نظر می گیرد، می تواند در هر بار اجرا اندکی متفاوت با مقادیری باشد که در ادامه گزارش می شود، این گزارش مبتنی بر آخرین اجرای این کد می باشد.

نتیجه: بهترین شبکه ی MLP به دست آمده دارای ویژگی های زیر می باشد.

۱. بهترین صحت طبقه بندی به دست آمده ( genetic\_mlp\_best\_acc ) = ۶۶.۴۶ درصد

۲. بهترین تعداد نوروں ها در لایه ی میانی ( genetic\_mlp\_best\_neuron\_size ) = ۳

۳. بهترین تابع فعالسازی ( genetic\_mlp\_best\_active\_func ) = 'tansig'

### بخش چهارم: طراحی شبکه ی RBF بر روی داده های آموزشی و یافتن بهترین آن

در این بخش دقیقاً مشابه و مطابق با جزییات و ویژگی هایی که در بخش 6 گزارش فاز ۱ بیان شد، عمل می کنیم با این تفاوت که در این بخش از ویژگی های genetic\_Train\_Features استفاده کرده و بهترین پارامترها را به دست می آوریم.

توضیح: این بهترین مقادیر صحت و ویژگی های به دست آمده با توجه به مقادیر اولیه ی رندمی که الگوریتم در هر بار در نظر می گیرد، می تواند در هر بار اجرا اندکی متفاوت با مقادیری باشد که در ادامه گزارش می شود، این گزارش مبتنی بر آخرین اجرای این کد می باشد:

نتیجه: بهترین شبکه ی RBF به دست آمده دارای ویژگی های زیر می باشد.

۱. بهترین صحت طبقه بندی به دست آمده ( genetic\_rbf\_best\_acc ) = ۶۶.۷۷ درصد

۲. بهترین مقدار برای شعاع نوروں های میانی ( genetic\_rbf\_best\_spread ) = ۳.۷

۳. بهترین مقدار برای حداکثر تعداد نوروں های میانی ( genetic\_rbf\_best\_N ) = ۵

مقایسه ی نتایج به دست آمده در این حالت با نتایج فاز ۱ :

میزان دقتی که بر روی داده های اعتبارسنجی به دست آمد در این حالت تقریباً برابر با میزان دقتی است که در فاز ۱ به دست آوردیم ؛ این اتفاق هم برای بهترین شبکه ی MLP و هم برای بهترین شبکه ی RBF افتاده است و به مانند بهترین شبکه های به دست آمده در فاز ۱ ، شبکه های نسبتاً ساده ای هستند.

### **بخش پنجم : طبقه بندی داده های تست با استفاده از بهترین شبکه ی MLP به دست آمده**

در این بخش ، با استفاده از بهترین پارامترهای به دست آمده در بخش سوم و هم چنین استفاده از داده های `genetic_Train_Features` و `genetic_Test_Features` به عنوان داده های آموزشی و تست ، داده های تست را طبقه بندی کرده و لیبل آن ها را به دست می آوریم ، این لیبل ها را تحت عنوان `genetic_mlp_predict_y_test.mat` ذخیره می کنیم .

### **بخش ششم : طبقه بندی داده های تست با استفاده از بهترین شبکه ی RBF به دست آمده**

در این بخش ، با استفاده از بهترین پارامترهای به دست آمده در بخش چهارم و هم چنین استفاده از داده های `genetic_Train_Features` و `genetic_Test_Features` به عنوان داده های آموزشی و تست ، داده های تست را طبقه بندی کرده و لیبل آن ها را به دست می آوریم ، این لیبل ها را تحت عنوان `genetic_rbf_predict_y_test.mat` ذخیره می کنیم .