

در این پروژه 4 فایل متلب با نام های زیر وجود دارند که هر کدام مربوط به شبکه عصبی با یک یا دو لایه پنهان به روش SGD و یا BGD می باشند

لازم به ذکر است محتویات این پروژه تکمیل کننده و اصلاح کننده پروژه یک نیز میباشد بنابراین محتویات پروژه یک را هم مرور میکنیم

- BGD\_1ShallowLayer
- BGD\_2ShallowLayer
- SGD\_1ShallowLayer
- SGD\_2ShallowLayer

بخش های مختلف کد های فایل های بالا در درون فایل ها کامنت گذاری شده اند که ساختار آن ها مشابه یکدیگر است و در ابتدای فایل متغیر های لازم، وزن ها و بایاس های اولیه ساخته می شوند و سپس محاسبات اصلی feedforward و backpropagation و سپس محاسبه خطا ها و رسم نمودار ها را داریم

بخش هسته ای کد به شکل زیر است که آن را در صفحه ی بعد توضیح میدهم

```
% initiate epochs
for k = 1 : epoch

    % feedforwarding all variables (BGD)
    % a(n) = w(n) * x + b
    result = weight1 * amoozesh + bias_list1(:,1:100);
    O1 = tansig(result);
    O2 = tansig(weight2 * O1 + bias_list2(:,1:100));
    O3 = weight3 * O2 + bias_list3(:,1:100);

    % calculating back propagation
    amoozesh_diff3 = amoozesh_data - O3;
    amoozesh_error(k)=sum(amoozesh_diff3*amoozesh_diff3', 2)/(2 * 100);
    weight3 = weight3 + eta * amoozesh_diff3 * O2';
    bias3 = bias3 + eta * amoozesh_diff3 * ones(100,1);
    bias_list3(1,:) = bias3(1,1);
    amoozesh_diff2 = weight3' * amoozesh_diff3;
    delta = amoozesh_diff2.* (4*exp(-2*result)./(1+exp(-2*result)).^2);
    weight2 = weight2 + eta * delta * O1';
    bias2 = bias2 + eta * amoozesh_diff2;
    for i = 1 : neuron_count
        bias_list2 (i,:) = bias2(i,1);
    end
end
```

در ابتدای این بخش یک حلقه به تعداد epoch های مورد نظر ساخته ایم که در بخش بالایی مقادیر اولیه را با فرمول نوشته شده feedforward میکند و سپس در مرحله بعد همزمان با محاسبه خطای آموزش طبق فرمول های مشتق های جزئی زنجیره ای backpropagation وزن و بایاس های جدید را ست میکنیم لازم به ذکر مقداری که در رابطه ی محاسبه ی دلتا میبینیم مقدار مشتق تابع سیگموید میباشد

## تفاوت SGD و BGD:

همانطور که در کد ها میبینید با توجه به اینکه در BGD همه ی مقادیر یکجا فید فروارد میشوند و محاسبات لازم انجام میشود اما در SGD این روند برای تک تک ورودی ها جدا گانه انجام میشود، در فایل های SGD به جای پاس دادن کل ماتریکس ورودی ها، با یک حلقه مقادیر ورودی تک به تک فید فروارد شده و وزن ها و بایاس ها با پس انتشار خطا آپدیت میشوند.

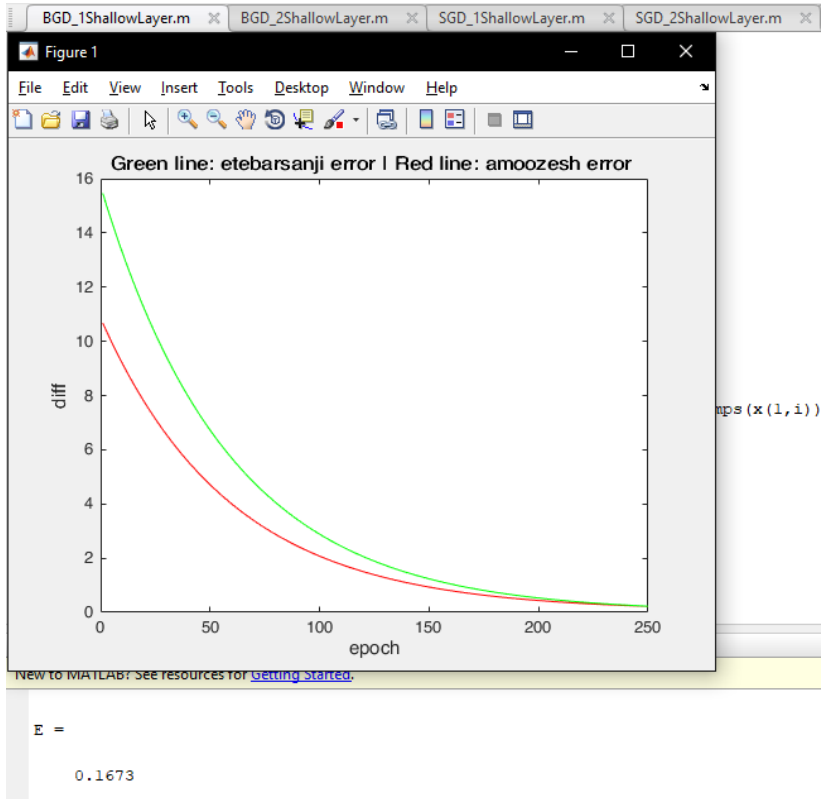
انتظار داریم در نمودار های خروجی شبکه های دو لایه از تک لایه خطای کمتری داشته باشند و خطای SGD از BGD بیشتر باشد

این نکته لازم به ذکر است که روش BGD بسیار کند بوده با این که نتیجه دقیق تری به ما میدهد اما استفاده از آن در هیچ منبعی توصیه نمیشود!!!

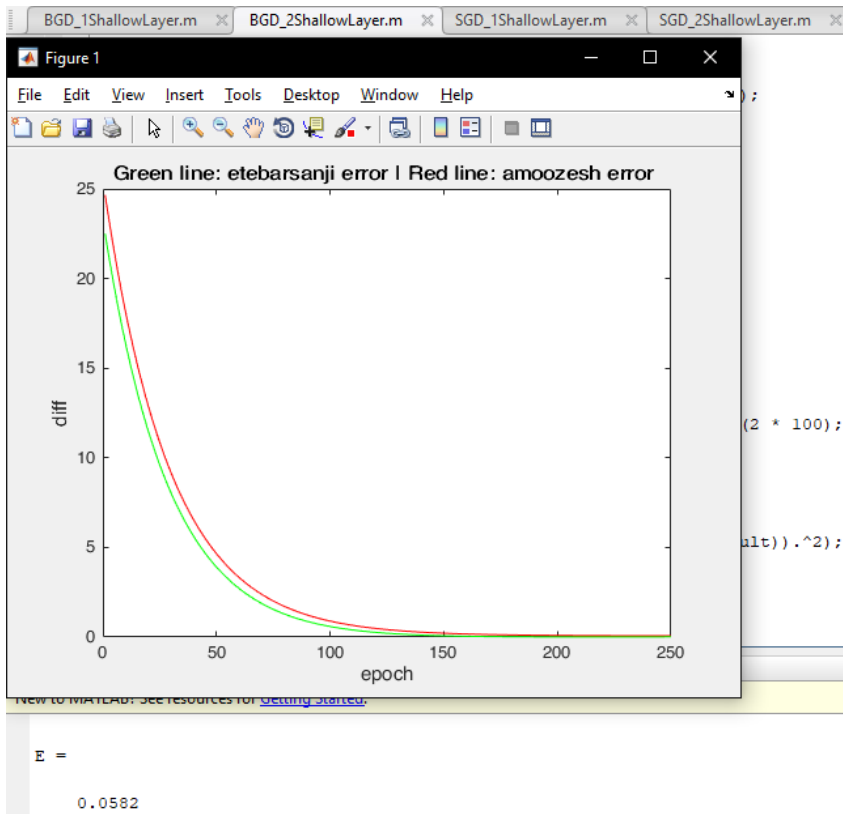
نمودار ها در صفحه ی بعد:

- خط سبز خطای اعتبار سنجی
- خط قرمز خطای آموزش
- مقدار E خطای نهایی می باشد

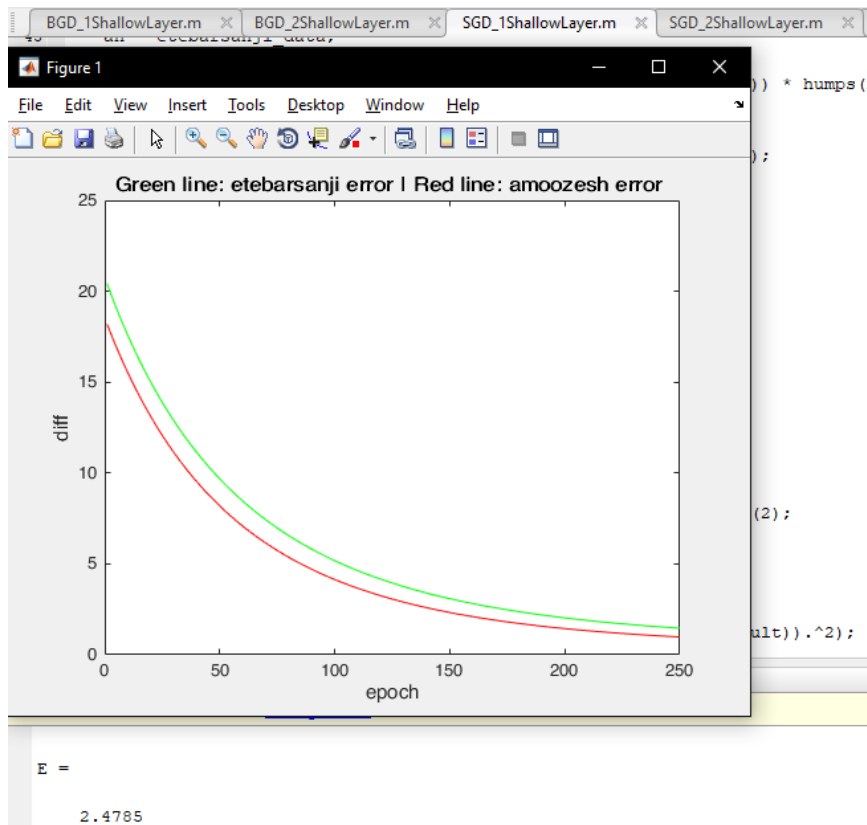
### • BGD\_1ShallowLayer: خطأ 0.16



• BGD\_2ShallowLayer: خطأ 0.058



## • SGD\_1ShallowLayer: خطا 2.478



## • SGD\_2ShallowLayer: خطا 1.43

