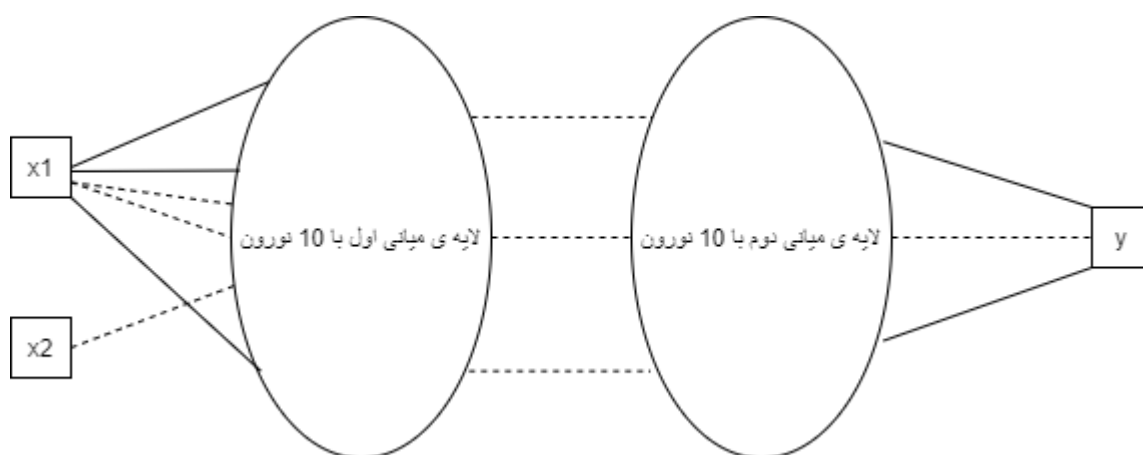


گزارش پروژه شبکه عصبی

سید آرش ساعتچی

9727283

طرح کلی:



توضیح کد :

در میان توابع تعریف شده ، اکثرا را از روی اسمشان میتوان تشخیص داد که چه کاربردی دارند و در اینجا به اختصار به توضیح برخی از آن ها میپردازیم:

تابع target_function :

این تابع همان تابع اصلی است که برنامه باید با استفاده از شبکه عصبی آنرا تخمین بزند.

تابع matrix_function :

توابع ریاضی که بالاتر تعریف شد ، همگی فقط میتوانند یک ورودی یا دو ورودی را تبدیل به یک خروجی کنند . ولی ما در کد نیازمند این هستیم که برنامه بتواند ماتریسی از ورودی را گرفته و با اعمال یک فانکشن روی تمام درایه های آن ، یک ماتریس خروجی با همان ابعاد به ما تحویل دهد . این تابع دقیقا همین کار را می کند و همانطور که میبینید ، ورودی های آن یک ماتریس و همچنین یک فانکشن است.

تابع calcute_E :

این تابع ، مقدار ارور های داخل یک آرایه یا ماتریس را به توان دو رسانده ، با هم جمع کرده و در اخر نصف می کند.

بعد از تعریف توابع مورد نیاز باید ورودی های آموزش ، اعتبارسنجی و تست خود را مقدار دهی کنیم .

لازم به ذکر است برای ساخت ورودی های آموزش ، یک ماتریس 30 در 30 انتخاب شده و با شافل کردن (shuffle) درایه های آن به 900 تا ورودی رندوم خود میرسیم. چون بازه ی اعداد انتخاب شده ، بین صفر و یک است ، نرمال کردن لازم نیست. سپس ماتریس های target را از ورودی هایی که بالاتر مقداردهی کردیم، میسازیم و نرمالایز (normalize) نیز میکنیم.

سپس وزن های اولیه خود را به صورت تصادفی میسازیم

`random.random()`

این تابع پایتون ، مقداری تصادفی از صفر تا یک به ما میدهد

در نهایت وارد حلقه ی اصلی برنامه شده و مقادیر feedforward و backpropagation را مطابق فرمول هایی که در کلاس تدریس شد ، بدست می آوریم . خطا های بدست آمده را در دو آرایه ی learning_errors , validation_errors ذخیره می کنیم و پس از پایان حلقه ، از آنها برای رسم نمودار استفاده میکنیم.

دقت کنید که در پایتون و نامپای برای انجام ضرب های ماتریسی به صورت زیر عمل می کنیم:

فرض کنیم A , B ماتریس های از پیش تعریف شده هستند

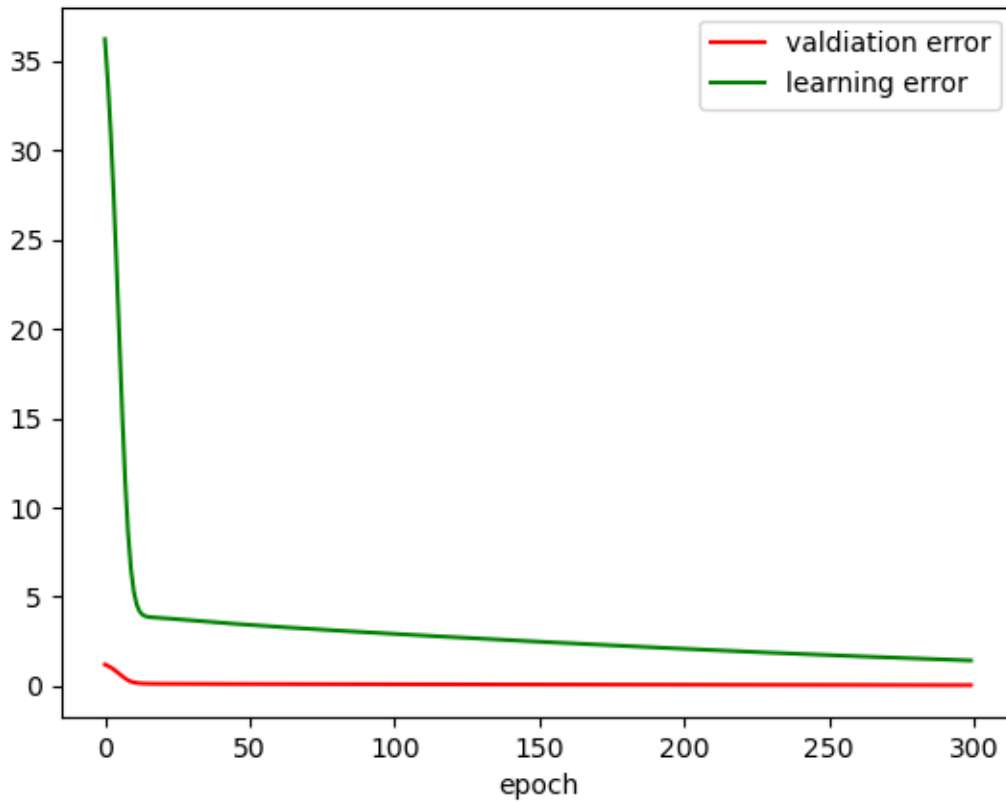
`A * B` -> Point multiplication

`A.dot(B)` -> Matrix multiplication

در ابتدای برنامه ماکرو هایی وجود دارد که میتوان با تغییر آنها ساختار شبکه عصبی را بهبود بخشید . مانند تعداد نرون ها یا تعداد داده های آموزش و اعتبارسنجی و یا تعداد epoch ها.

نتایج زیر برای شبکه ی عصبی با epoch 300 را مشاهده می کنیم:

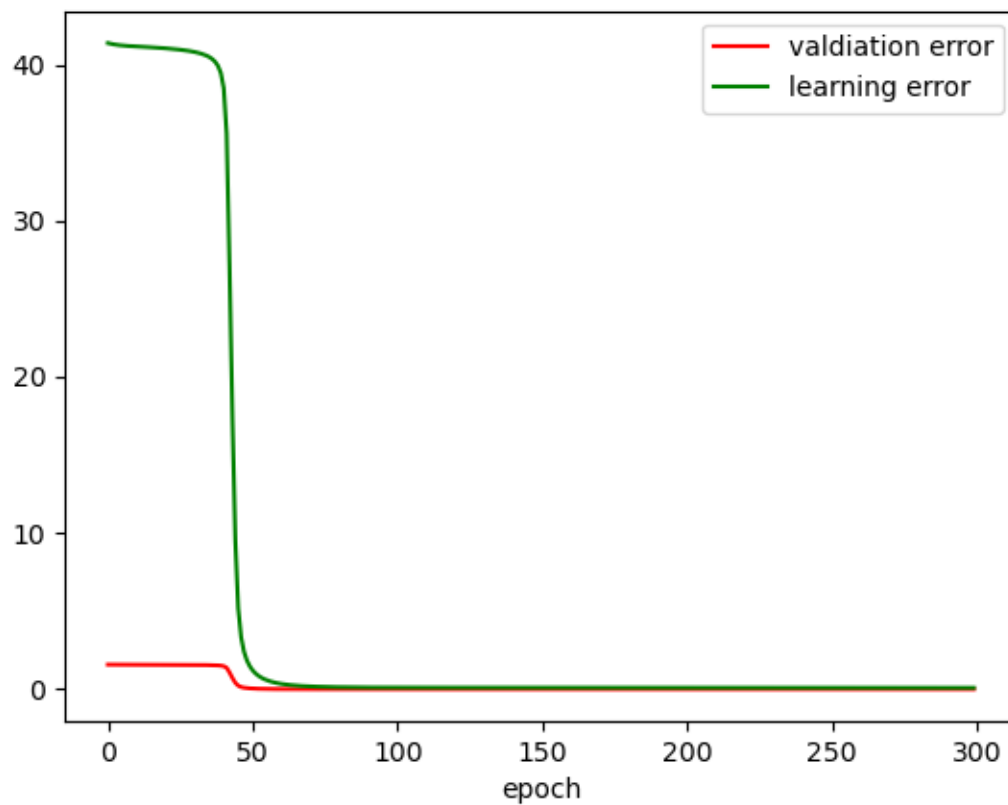
شبکه عصبی با یک لایه ی میانی:



همچنین خطاهای مربوط به داده های تست بدست آمده:

[[0.08961638 , 0.07324588 , 0.05691098 , 0.04067202 , 0.02460008 , 0.00877874
-0.0066939 , -0.02170286 , -0.03611359 , -0.04976896]]

شبکه عصبی با دو لایه ی میانی:



همچنین خطاهای مربوط به داده های تست بدست آمده:

```
[[ 0.01766339  0.01443846  0.01113416  0.00775653  0.00431582  0.00082757  
-0.00268595 -0.00619412 -0.00965602 -0.01301805]]
```