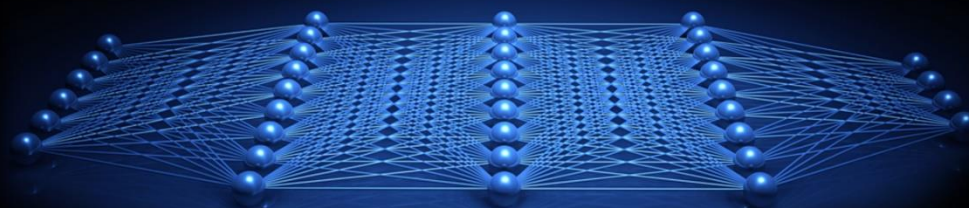


## پروژه اول درس یادگیری عمیق

دکتر سید ابوالقاسم میرروشندل



طراح: مهسا حسینی

### قبولی یا رد دانش آموزان

#### داستان پروژه:

زهره به مدت ۵ سال است که در دبیرستان برف، تدریس می کند. همه چیز عادی پیش میرفت تا اینکه بیماری جدیدی به نام امیکرون ۲۲ شیوع یافت و دبیرستان ها غیر حضوری شد. برای مدیریت شرایط تحصیلی، قانونی مبنی بر این اعلام شد که اولاً صرفاً نمره قبولی یا رد برای دانش آموزان در نظر گرفته شود و ثانیاً هر دبیر می تواند در روزها و ساعت های متفاوت از هم برای گروه های متفاوت (که هر گروه دارای تعداد کمی از دانش آموزان است)، جهت امتحان حضوری برنامه ریزی کند. ولی زهره میخواهد که تا حد امکان سلامت دانش آموزان به خطر نیفتد و از طرفی هم در مورد نمره دادن میخواهد به هر دانش آموز نمره ای بدهد که تا حد ممکن حق کسی ضایع نشود. او تصمیم می گیرد دو روش نمره دهی در نظر بگیرد؛ به این صورت که به دانش آموزانی که در آن درس و در طول آن ترم دارای ویژگی های خاصی هستند، نمره بالا بدهد (قبول می شوند) و برای دانش آموزانی که طبق این روند، نمره پایینی می گیرند (رد می شوند)، فرصتی جهت جبران بدهد و از آنها امتحان حضوری بگیرد. زهره از شما میخواهد به او کمک کنید با استفاده از پیاده سازی شبکه عصبی مدلی بسازد که تا حد امکان پیش بینی دقیقی داشته باشد زیرا سلامت دانش آموزانش و همچنین ضایع نشدن حق آنها دو اولویت مهم او است (جهت آموزش و سپس بررسی دقت مدل، او داده های ترم های حضوری را در اختیار دارد).

## هدف پروژه:

هدف این پروژه، استفاده از ساختار شبکه عصبی از آغاز (Neural Networks from Scratch) به منظور پیش بینی قبولی یا رد دانش آموزان با توجه به مجموعه ای از ویژگی های آنها در طول یک ترم تحصیلی است.

## شرح پروژه:

فرآیند کلی به صورت زیر است:

۱. داده های ورودی را به شبکه عصبی تغذیه می کنیم.
۲. داده ها از لایه ای به لایه دیگر جریان می یابد تا زمانی که خروجی نهایی آماده شود.
۳. وقتی خروجی آماده شد، می توانیم خطا را محاسبه کنیم.
۴. در نهایت می توانیم یک پارامتر (وزن یا بایاس) را با کم کردن مشتق خطا نسبت به خود پارامتر تنظیم کنیم.
۵. سپس همین فرآیند را تکرار می کنیم.

به طور واضح تر، این پروژه شامل سه مرحله کلی است:

۱. پیاده سازی شبکه عصبی
۲. آموزش مدل پیاده سازی شده
۳. ارزیابی مدل آموزش دیده

توجه کنید عملیات بارگذاری و پیش پردازش داده های مورد نظر در فایل [prepare\\_data.ipynb](#) قرار گرفته است و داده ها به دو دسته داده های آموزشی (Training Set)، داده های ارزیابی (Validation Set) تقسیم شده اند که بر اساس آن شما بتوانید تمام تمرکزتان را روی پیاده سازی مدل شبکه عصبی بگذارید و درگیر مرحله پیش پردازش داده ها نشوید.

### ۱) پیاده سازی شبکه عصبی

در این بخش، به پیاده سازی شبکه عصبی برای دسته بندی دو کلاسه (قبول یا رد) می پردازیم. پیاده سازی این الگوریتم در حالت کلی از چهار مرحله تشکیل شده است:

۱. مقداردهی وزن ها و آستانه فعالسازی
۲. انتشار رو به جلو (Forward Propagation)
۳. انتشار رو به عقب (Backward Propagation)
۴. ترکیب مراحل پیشین و ساخت الگوریتم نهایی

## ۱. مقداردهی وزن ها و آستانه فعالسازی

در مرحله اول باید به تمامی مقادیر وزن ها و آستانه مقدار اولیه رندوم داده شود. وزن ها در ماتریسی به نام  $W$  و مقدار آستانه در متغیر عددی  $b$  قرار دارد.

## ۲. انتشار رو به جلو (Forward Propagation)

در این مرحله، داده های ورودی ( $X$ ) را به شبکه می دهیم، سپس با مقایسه نتیجه شبکه ( $A$ ) با خروجی مورد نظر ( $Y$ )، می توانیم خطا ( $E$ ) را محاسبه کنیم.

## ۳. انتشار رو به عقب (Backward Propagation)

هدف از این مرحله این است که خطا ( $E$ ) را به حداقل برسانیم.

## ۴. ترکیب مراحل پیشین و ساخت الگوریتم نهایی

در این مرحله هدف ترکیب توابع پیاده سازی شده در مراحل پیشین و ساخت مدل نهایی است. در این مرحله باید به تعداد  $k$  بار مراحل زیر را به ترتیب انجام دهید:

- ابتدا Forward Propagation را انجام داده و مقادیر  $A$  و  $E$  را ذخیره کنید.
- در گام بعدی Backward Propagation را انجام داده و مقادیر  $\frac{\partial E}{\partial b}$  و  $\frac{\partial E}{\partial W}$  را ذخیره کنید.
- در آخر مقادیر  $W$  و  $b$  را بروز رسانی کنید.

## ۲) آموزش مدل پیاده سازی شده

در این بخش باید الگوریتم پیاده سازی شده در بخش قبلی را روی داده های آموزشی که در اختیار شما قرار گرفته است، آموزش دهید (مقدار  $k$  پیشنهادی: ۱۰۰۰، مقدار نرخ یادگیری ( $\alpha$ ) پیشنهادی: ۰/۱). همچنین لازم است در هر  $\frac{k}{10}$ ، دقت (Accuracy) و Loss آن روی داده های آموزشی، محاسبه شود.

## ۳) ارزیابی مدل آموزش دیده

در مرحله آخر و پس از آموزش مدل، برای ارزیابی عملکرد مدل، باید Accuracy و Loss آن روی داده های ارزیابی، محاسبه شود.

فرمول ها و توضیحات نمادهای به کار رفته در آنها:

$$Z = W^T X + b$$

$$A = \text{sigmoid}(Z)$$

$$E = \frac{-1}{n} \sum_{i=1}^n (y^i \log(a^i) + (1 - y^i) \log(1 - a^i))$$

$$\frac{\partial E}{\partial W} = \frac{1}{n} X (A - Y)^T$$

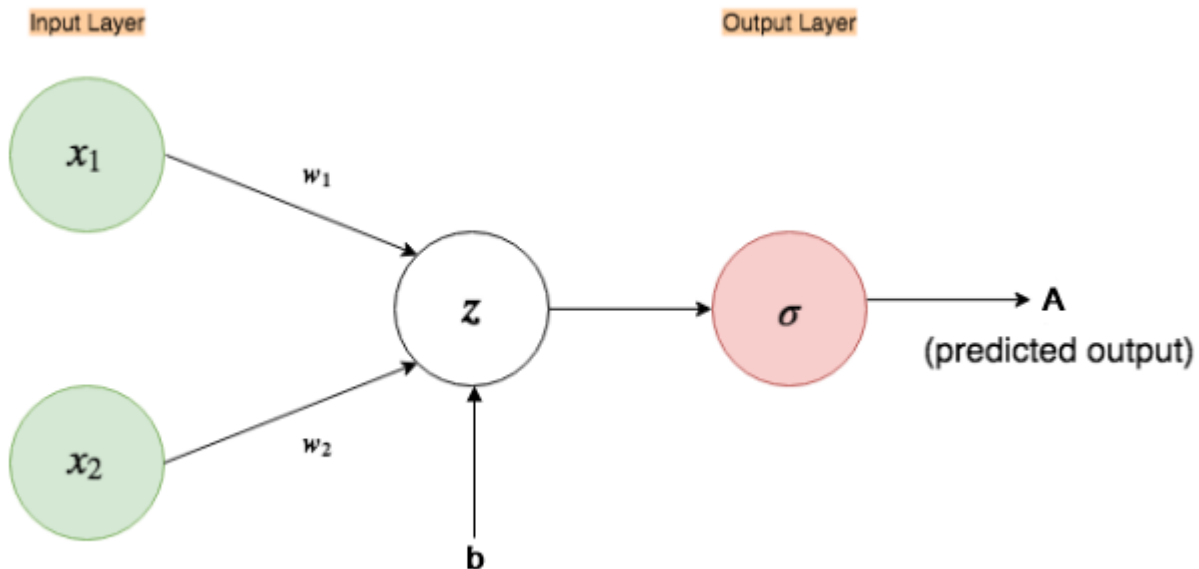
$$\frac{\partial E}{\partial b} = \frac{1}{n} \sum_{i=1}^n (a^i - y^i)$$

$$W = W - \alpha \frac{\partial E}{\partial W}$$

$$b = b - \alpha \frac{\partial E}{\partial b}$$

- X ماتریسی از ویژگی‌های عضوهای داده‌های آموزشی است که در هر ستون آن ویژگی‌های یک عضو قرار دارد.
- A آرایه‌ای از برچسب‌های پیش‌بینی شده برای هر عضو داده‌های آموزشی را شامل می‌شود و  $a^i$  نشانگر عضو iام از این آرایه است (به عبارت دیگر  $a^i$  نشانگر مقدار پیش‌بینی شده برای عضو iام از داده‌های آموزشی است).
- همچنین  $y^i$  نیز نشانگر مقدار برچسب واقعی عضو iام است.
- متغیر n نیز نشانگر تعداد کل داده‌های آموزشی است.

جهت درک بهتر موضوع، شکل زیر به عنوان نمونه ارائه می‌شود که دارای دو لایه (لایه input و لایه output)، یک آستانه و دو ورودی ( $x_1$  و  $x_2$ ) و خروجی پیش‌بینی شده توسط شبکه (A) است. همچنین برای لایه output از تابع فعال‌سازی sigmoid استفاده شده است.



## نکات تکمیلی:

- در مرحله ی پیش پردازش مقادیر کیفی تمام ویژگی ها و مقادیر کیفی نمرات به مقادیر عددی نگاشت شده است تا بتوانید داده ها را به راحتی آموزش داده و ارزیابی کنید.
- برای لایه output از تابع فعال سازی sigmoid استفاده کنید.
- توضیحاتی که ارائه شد، برای تعداد لایه ۲ بود. روند ذکر شده را برای تعداد لایه های ۳ و ۵ نیز تکرار کنید و نتیجه تعداد لایه های ۲ و ۳ و ۵ را در جدول موجود در فایل گزارش وارد کنید.
- منظور از تعداد لایه های ۲ و ۳ و ۵، مجموع تعداد لایه های hidden و لایه input و output است.
- اتصالات بین لایه ها را fully connected در نظر بگیرید.
- پروژه را یکبار دیگر انجام دهید با این تفاوت که در مرحله اول به تمامی مقادیر وزن ها و آستانه مقدار اولیه صفر داده شود.
- برای آموزش و ارزیابی مدل، دادگان به نام [Student's Academic Performance](#) در فایل Dataset.csv در اختیار شما قرار داده شده است.
- در دیتاست داده شده، ستون class نشان دهنده نمره است که دارای مقدار H یا M یا L است.
- نمره H و M را قبولی (به صورت عددی: ۱) و نمره L را رد (به صورت عددی: ۰) در نظر بگیرید.
- برچسب مورد نظر باید ۱ یا ۰ باشد، بدین منظور در صورت دادن خروجی پیوسته در مرحله محاسبه دقت، میتوانید از روش انتخاب یک آستانه و در نتیجه تبدیل عددها به ۱ و ۰ استفاده کنید.
- به میزان بهبود دقت نسبت به میانگین دقت بدست آمده توسط تمامی گروه ها، نمره مثبت بیشتری برای شما در نظر گرفته می شود. بهبود دقت را میتوانید از طریق ایجاد تغییراتی در مرحله پیش پردازش و یا تغییر  $\alpha$  و  $k$  بدست بیاورید.
- با وجود داده شدن فرمول ها، در زمان تحویل پروژه باید توانایی محاسبه و توضیح آنها را داشته باشید.
- گزارش پروژه بایستی کامل بوده و دقیق به سوالات و بخش های مربوطه پاسخ داده باشد.
- پروژه دانشجویی نمودارهای تغییر Loss و Accuracy هر دو مجموعه آموزش و ارزیابی (تعداد لایه های ۲ و ۳ و ۵) را داشته باشد.
- پیاده سازی به صورت تک نفره است و هیچ محدودیتی برای زبان برنامه نویسی وجود ندارد.
- در فرآیند پیاده سازی، مجاز به استفاده از هیچ فریمورک آماده یادگیری عمیق نیستید بلکه باید با محاسبات و کدهای ابتدایی شبکه عصبی درگیر شوید.
- برای دستیابی به اطلاعات بیشتر در مورد شبکه عصبی، می توانید به این [لینک](#) مراجعه کنید.
- بحث و بررسی میان دانشجویان آزاد است اما هر دانشجو موظف است پروژه را به تنهایی انجام دهد و در هنگام تحویل حضوری، به تمام جزئیات کد کاملاً مسلط باشد. با موارد **تقلب** و **کپی کردن**، طبق تشخیص دوستان حل تمرین، برخورد جدی خواهد شد.
- با توجه به وجود کدهای آماده در دسترس، تشابه کد هر دانشجو با آن کدها نیز بررسی می شود و در صورت تشابه، به عنوان **تقلب** و **کپی کردن**، تلقی می شود.
- توجه کنید که کدهای شما باید خوانا و دارای کامنت گذاری مناسب باشد.
- زمان بندی و چگونگی تحویل حضوری پروژه، متعاقباً اعلام خواهد شد.