



## مقدمه

هدف این تمرین، آشنایی شما با شبکه‌های عصبی عمیق می‌باشد. در این پروژه، شما باید از ابتدا هر بخش کوچک از شبکه عصبی را پیاده‌سازی کرده و عملکرد و خروجی آن را با توابع دیگری که تعریف شده است بررسی کنید. در انتها، شما بایستی این بخش‌ها یا لایه‌ها را کنار هم بچینید تا به یک شبکه عصبی عمیق دست پیدا کنید. در نهایت از این شبکه برای دو کار طبقه‌بندی و رگرسیون باید استفاده کنید و آن را روی داده‌های MNIST و California Housing Dataset آموزش و اعتبارسنجی نمایید.

## مرحله اول: پیاده‌سازی هر یک از لایه‌های یک شبکه عصبی

در این مرحله، شما با چندین لایه شبکه عصبی مواجه می‌شوید که هر یک بخشی برای تکمیل دارد. این لایه‌ها عبارتند از:

1. لایه Fully Connected
2. تابع فعال‌ساز Relu
3. تابع فعال‌ساز Sigmoid
4. لایه ساندویچی (با کنار هم گذاشتن یک لایه Fully Connected با یک تابع فعال‌ساز Relu)
5. تابع هزینه Softmax
6. تابع هزینه MSE

هر یک از این لایه‌ها شامل پارامترهایی هستند که حین آموزش به‌روزرسانی می‌شوند. این به‌روزرسانی به صورت دو مرحله Forward Pass و Backward Pass انجام می‌شود. در مرحله Forward Pass لایه‌ها با گرفتن ورودی و داشتن پارامترهای مختص به آن لحظه، خروجی تولید می‌کنند. در مرحله بعدی یعنی Backward Pass، گرادیان‌ها محاسبه می‌شوند و بعداً با توجه به گرادیان‌ها، مقادیر قبلی پارامترها و سیاست بهینه‌سازی، پارامترهای آن لایه به‌روزرسانی می‌شوند.

- لایه Fully Connected:

□ پیاده‌سازی Forward Pass:

در این قسمت باید تابع affine forward را که وزن های  $w$  و  $b$  و ورودی  $x$  را دریافت می‌کند تکمیل کنید. سپس خروجی این تابع را با ماتریس out\_correct مقایسه کنید. اختلاف این دو باید ناچیز باشد.

□ پیاده‌سازی Backward Pass:

حال باید affine backward را تکمیل کنید و خروجی آن را با خروجی تابع گرادیان عددی مقایسه کنید. اختلاف این دو باید ناچیز باشد.

- تابع فعال‌ساز Relu

□ پیاده‌سازی Forward Pass:

تابع Relu\_forward را تکمیل کنید و خروجی آن را با ماتریس out\_correct مقایسه کنید. اختلاف این دو باید ناچیز باشد.

□ پیاده‌سازی Backward Pass:

حال باید Relu\_backward را تکمیل کنید و خروجی آن را با خروجی تابع گرادیان عددی مقایسه کنید. اختلاف این دو باید ناچیز باشد.

- تابع فعال‌ساز Sigmoid

□ پیاده‌سازی Forward Pass:

تابع Sigmoid\_forward را تکمیل کنید و خروجی آن را با ماتریس out\_correct مقایسه کنید. اختلاف این دو باید ناچیز باشد.

□ پیاده‌سازی Backward Pass:

حال باید Sigmoid\_backward را تکمیل کنید و خروجی آن را با خروجی تابع گرادیان عددی مقایسه کنید. اختلاف این دو باید ناچیز باشد.

- لایه ساندویچی:

این لایه از کنار هم گذاشتن لایه Fully Connected با تابع فعال‌ساز Relu به دست می‌آید.

□ پیاده‌سازی Forward Pass و Backward Pass:

حال باید affine\_relu\_forward و affine\_relu\_backward را تکمیل کنید و خروجی

تابع affine\_relu\_backward را با خروجی تابع گرادیان عددی مقایسه کنید. اختلاف این دو باید ناچیز باشد.

- توابع هزینه Softmax و MSE:

توابع softmax\_loss و MSE\_loss را تکمیل کرده و از صحت خروجی آن‌ها با مقایسه با خروجی تابع گرادیان عددی اطمینان حاصل کنید. تفاوت باید ناچیز باشد.

بعد از پیاده‌سازی موارد ذکر شده فوق، به این سوالات پاسخ مختصر دهید:

1. توابع فعال‌ساز به چه منظور استفاده می‌شوند؟ دو مورد از آن‌ها را نام ببرید و نحوه عملکرد آن‌ها را بیان کنید.
2. نقش تابع هزینه در آموزش مدل‌ها چیست؟ دو مورد از آن‌ها را نام ببرید و نحوه عملکرد آن‌ها را بیان کنید. هر یک از این توابع در چه شرایطی مورد استفاده قرار می‌گیرند؟
3. Batch Normalization چیست و به چه منظور استفاده می‌شود؟
4. چه زمانی شبکه‌های عصبی نسبت به الگوریتم‌های یادگیری ماشین کلاسیک برتری دارند؟
5. اگر بخواهیم یک شبکه عصبی با ۱۰۰۰ پارامتر را تنها با ۱۰ داده آموزش دهیم، چه اتفاقی خواهد افتاد؟

## مرحله دوم: کنار هم گذاشتن این بخش‌ها در کنار هم

بعد از اینکه مطمئن شدید تمام توابع درست کار می‌کنند. وقت این است که با چیدن این لایه‌ها در کنار هم به یک معماری شبکه عصبی دست پیدا کنیم. در این مرحله شما باید تابع FullyConnectedNet را تکمیل کنید.

Stochastic Gradient descent یا SGD با استفاده از گرادیان، پارامترهای مدل را در جهت کاهش تابع هزینه، بروزرسانی می‌کند. Momentum باعث کاهش نوسانات در حرکت در جهت کاهش تابع هزینه می‌شود. استفاده همزمان از SGD و Momentum باعث کاهش نوسانات و افزایش سرعت همگرایی به سمت مینیمم تابع هزینه می‌شود. با توجه به توضیحات داده شده، تابع sgd\_momentum را پیاده‌سازی کنید.

6. خلاصه‌ای از چگونگی عملکرد SGD و Momentum ارائه دهید.

## مرحله سوم: آموزش و اعتبارسنجی شبکه عصبی ایجاد شده

MNIST یک مجموعه داده مشهور شامل تصاویر دست‌نویس ارقام 0 تا 9 است. هر تصویر سیاه و سفید، با ابعاد  $28 \times 28$  پیکسل است و به صورت اعداد دیجیتالی بینایی کامپیوتر و یادگیری ماشین به کار می‌رود. در اینجا شما با استفاده از توابعی که تا به اینجا پیاده‌سازی کرده‌اید و همچنین توابع کلاس solver که در اختیار شما قرار گرفته، برای دسته‌بندی این تصاویر، بهترین مدلی که می‌توانید را روی این مجموعه داده آموزش داده و ارزیابی کنید. نتایج خود را نیز تحلیل کنید.

در ادامه از شما خواسته می‌شود تا با استفاده از مجموعه دادگان California Housing dataset، با آموزش بهترین مدل ممکن، قیمت خانه‌ها را پیش‌بینی کنید. (اختیاری)

## مرحله چهارم: آموزش و اعتبارسنجی شبکه عصبی پیچشی

در کلاس درس، شما با شبکه‌های عصبی کانولوشنال آشنا شدید. در این مرحله، شما با آموزش یک شبکه عصبی پیچشی بر روی مجموعه دادگان CIFAR 10، جزئیات این شبکه‌ها را نیز می‌آموزید. برای سادگی کار، از شما خواسته شده تا از مدل VGG16 استفاده کنید و نیازی به طراحی شبکه جدیدی نیست. جزئیات این مدل در نوت‌بوکی که در اختیار شما قرار گرفته است، توضیح داده شده است. در فایل نوت‌بوکی که در اختیار شما قرار گرفته است، تعدادی از سلول‌ها خالی می‌باشد که شما می‌بایست با توجه به توضیحات بالای آن سلول، کدهای خواسته شده را پیاده‌سازی کنید.

7. نقش لایه‌های پیچشی در این شبکه‌ها چیست؟ چرا پس از آن‌ها لایه‌های تماماً متصل قرار می‌گیرد؟

8. توضیح دهید چرا برای استخراج ویژگی از لایه‌های تماماً متصل استفاده نمی‌شود؟

## نکات پایانی

- دقت کنید که کد شما باید به نحوی زده شده باشد که نتایج قابلیت بازتولید داشته باشند.
- توضیحات مربوط به هر بخش از پروژه را بطور خلاصه و در عین حال مفید در گزارش خود ذکر کنید. از ابزارهای تحلیل داده مانند نمودارها استفاده کنید. حجم توضیحات گزارش شما هیچ گونه تاثیری در نمره نخواهد داشت و تحلیل و نمودارهای شما بیشترین ارزش را دارد.
- درباره هر بخش از مراحل پروژه می‌بایست علل استفاده یا عدم استفاده از هر الگوریتم، مزایا و معایب، عملکرد، فرا پارامترها و وضعیت خروجی‌ها را بطور دقیق مطالعه کنید. از این موضوعات در زمان تحویل پرسیده خواهد شد.
- از نحوه عملکرد کدهایی که در اختیار شما قرار گرفته شده در حین تحویل پرسیده خواهد شد.
- سعی کنید از پاسخ‌های روشن در گزارش خود استفاده کنید و اگر پیش‌فرضی در حل سوال در ذهن خود دارید، حتما در گزارش خود آن را ذکر نمایید.
- پس از مطالعه کامل و دقیق صورت پروژه، در صورت وجود هرگونه ابهام یا سوال با طراحان پروژه در ارتباط باشید.
- نتایج، گزارش و کدهای خود را در قالب یک فایل فشرده با فرمت AI\_CA4\_[stdNumber].zip در سامانه ایلرن بارگذاری کنید. به طور مثال AI\_CA4\_810101999.zip
- محتویات پوشه باید شامل فایل پاسخ‌های شما به سوالات کتبی، فایل jupyter-notebook، خروجی html و فایل‌های مورد نیاز برای اجرای آن باشد. از نمایش درست خروجی‌های مورد نیاز در فایل html مطمئن شوید.
- توجه کنید این تمرین باید به صورت تک‌نفره انجام شود و پاسخ‌های ارائه شده باید نتیجه فعالیت فرد نویسنده باشد. در صورت مشاهده تقلب به همه افراد مشارکت‌کننده، نمره تمرین 100- و به استاد نیز گزارش می‌گردد. همچنین نوشته نشدن کدها توسط هوش مصنوعی نیز بررسی می‌شود!