



به نام خداوند جان و خرد

طراحان: علی هدایت‌نیا، فرشاد سنگری



یادگیری عمیق با کاربردهایی در بینایی ماشین و پردازش صوت

تمرین اول

اسفند ماه ۱۴۰۱

فهرست

5	مقدمه - شبکه عصبی پیشرو.....
5	شبکه‌های عصبی پیشرو.....
6	سوال ۱ - داده‌های جدولی.....
6	معرفی داده.....
7	نمایش داده‌ها.....
7	تعریف مسئله.....
7	سوال‌ها.....
8	سوال ۱.۱.....
9	سوال ۱.۲.....
9	سوال ۱.۳.....
9	سوال ۱.۴.....
9	سوال ۱.۵.....
9	سوال ۱.۶.....
10	سوال ۱.۷.....
10	سوال ۱.۸ (امتیازی).....
10	سوال ۱.۹.....
10	سوال ۱.۱۰.....
10	سوال ۱.۱۱.....
10	سوال ۱.۱۲ (امتیازی).....
10	سوال ۱.۱۳ (امتیازی).....

- سوال ۲ - مسأله رگرسیون 11
- معرفی داده 11
- تعریف مساله 11
- سوال ها 12
- سوال ۲.۱ 12
- سوال ۲.۲ 12
- سوال ۲.۳ 12
- نکات تحویل 13
- پیاده سازی 13
- گزارش 13
- بارگذاری 14
- ارتباط با ما 14

فهرست جدول‌ها

- جدول ۱- جدول پارامترهای شبکه عصبی 8
- جدول ۲- ساختار پروژه 15

مقدمه - شبکه عصبی پیشرو

شبکه‌های عصبی پیشرو

شبکه‌های عصبی دسته‌ای از الگوریتم‌ها هستند که در یادگیری عمیق، زیرمجموعه‌ای از یادگیری ماشین محسوب می‌شوند. آنها از ساختار و عملکرد مغز انسان الهام گرفته شده‌اند و برای مدل سازی الگوها و روابط پیچیده در داده‌ها استفاده می‌شوند.

یک شبکه عصبی از لایه‌هایی از «نورون‌ها» تشکیل شده است که توابع ریاضی ای هستند که ورودی‌ها را می‌گیرند، محاسبات را انجام می‌دهند و نتیجه را به دست می‌آورند. نورون‌ها در یک شبکه عصبی از طریق لایه‌ها سازماندهی شده‌اند. به طوریکه هر لایه از لایه قبلی تغذیه می‌کند. لایه اول معمولاً لایه ورودی است که در آن داده‌ها به شبکه وارد می‌شود. آخرین لایه خروجی است که نتیجه نهایی محاسبات شبکه را تولید می‌کند.

در بین لایه‌های ورودی و خروجی، یک یا چند لایه "مخفی" وجود دارد. این لایه‌های پنهان به شبکه اجازه می‌دهند تا روابط پیچیده‌تری را بین ورودی‌ها و خروجی‌ها مدل سازی کند، زیرا آنها می‌توانند نمایش ویژگی‌های میانی و نمایش داده‌ها را یاد بگیرند.

یک شبکه عصبی در طول آموزش مقادیر زیادی از داده‌ها را تغذیه می‌کند و پارامترهای آن (مانند وزن و بایاس نورون‌ها) به طور مکرر تنظیم می‌شوند تا خطا بین خروجی پیش بینی شده و خروجی واقعی به حداقل برسد. این فرآیند به عنوان پس انتشار شناخته می‌شود و به شبکه اجازه می‌دهد تا در طول زمان پیش بینی‌های بهتری انجام دهد. بهبود یابد.

یادگیری عمیق به استفاده از شبکه‌های عصبی با لایه‌های متعدد (یعنی معماری‌های عمیق) برای مدل سازی الگوها و روابط پیچیده در داده‌ها اشاره دارد. شبکه‌های عصبی عمیق قادر به یادگیری و نمایش روابط بسیار غیرخطی و سلسله مراتبی در داده‌ها هستند و برای دستیابی به عملکرد پیشرفته در طیف گسترده‌ای از وظایف، از جمله تشخیص تصویر، تشخیص گفتار، پردازش زبان طبیعی و ... به کار می‌روند.

سوال ۱ – داده‌های جدولی

معرفی داده

CIFAR-10 یک مجموعه داده طبقه بندی تصویر پر کاربرد در زمینه یادگیری ماشین و بینایی کامپیوتر است. مجموعه داده شامل 60000 تصویر رنگی 32×32 در 10 کلاس با 6000 تصویر در هر کلاس است. کلاس ها شامل موارد زیر می باشد:

1. هواپیما
2. خودرو
3. پرنده
4. گربه
5. آهو
6. سگ
7. قورباغه
8. اسب
9. کشتی
10. کامیون

مجموعه داده به 50000 تصویر آموزشی و 10000 تصویر آزمایشی تقسیم شده است. تصاویر موجود در مجموعه داده نسبتاً با وضوح پایین و بسیار متنوع هستند که آنها را برای کارهای طبقه بندی تصاویر چالش برانگیز می کند.

مجموعه داده CIFAR-10 برای اولین بار در سال 2009 به عنوان زیرمجموعه ای از مجموعه داده های بزرگتر CIFAR-100 معرفی شد. این به عنوان یک مجموعه داده معیار برای الگوریتم های طبقه بندی تصویر طراحی شده است و از آن زمان به یک مجموعه داده استاندارد برای آزمایش مدل های جدید یادگیری ماشین تبدیل شده است.

مجموعه داده اغلب برای ارزیابی عملکرد شبکه های عصبی کانولوشن (CNN) در وظایف طبقه بندی تصویر استفاده می شود. بسیاری از مدل های پیشرفته، از جمله VGG، AlexNet، و ResNet، بر روی مجموعه داده CIFAR-10 آموزش و ارزیابی شده اند.

یک نکته مهم در مورد مجموعه داده CIFAR-10 این است که در مقایسه با سایر مجموعه داده های طبقه بندی تصویر محبوب مانند ImageNet نسبتاً کوچک است. با این حال، این می تواند یک مزیت برای محققانی باشد که می خواهند به سرعت مدل های جدید یادگیری ماشینی را نمونه سازی و آزمایش کنند.

نمایش داده ها

از هر کلاس به صورت رندم ۱۰ نمونه را به صورت رندم انتخاب کرده و نمایش دهید. (خروجی بایستی یک نمودار 10×10 باشد که هر ردیف مربوط به یکی از کلاس هاست.)

تعریف مسئله

در این تمرین قصد داریم با پیاده سازی شبکه های عصبی از صفر، با چگونگی عملکرد این شبکه ها به طور کامل آشنا شویم. برای تعریف و آموزش شبکه عصبی خواسته شده در هر قسمت شایسته است که پارامترهای مربوطه را در یک فایل `*.yaml` قرار دهید و از آن ها در آن قسمت استفاده کنید. ذکر مجدد این نکته خالی از لطف نیست که اعمال تغییرات خواسته شده باید به سادگی مقدور باشد. به طور مثال، بتوان تعداد لایه های شبکه عصبی را تغییر داد.

توجه شود که در هر سوال بایستی نمودار دقت، خطا و ماتریس در هم ریختگی برای دادگان آموزش و ارزیابی رسم شود.

مدل با پارامترهای اولیه ذکر شده در جدول 1 را به عنوان مدل پایه در نظر گرفته و در هر سوال تنها پارامتری که در سوال مطرح شده است را تغییر داده و سایر پارامترها را تغییر ندهید.

سوال ها

در سوال ۱.۱ یک شبکه عصبی را به طور کامل از ابتدا پیاده سازی می کنیم و در سوال ۱.۲، این شبکه عصبی را به کمک پارامترهای موجود در جدول (۱) آموزش می دهیم. در سایر سوالات سعی می کنیم که بررسی پارامترهای مختلف بپردازیم.

جدول ۱- جدول پارامترهای شبکه عصبی

پارامتر	مقادیر مختلف تعریف شده	مقدار اولیه
توزیع اولیه وزن ها	توزیع نرمال	$\mu = 0$ $\sigma = 1$ Bias = 0
تعداد ایپاک	[1,...]	50
نوع ورودی	Normalized, Standardized, Do Nothing	نرمالایز شده
اندازه بسته	[1,...]	32
نرخ یادگیری	$0 < lr$	1e-3
تعداد لایه ها و نورون ها	حداقل: یک لایه با یک نورون تا چندین لایه با نورون های دلخواه	یک لایه با 16 نورون
مومنتوم	وجود یا عدم وجود مومنتوم	عدم وجود
تابع فعالساز	Sigmoid, tanh, ReLu, LeakyReLu	ReLu
تابع هزینه	MSE, Cross-Entropy	Cross-Entropy

سوال (۱.۱)

روابط پیاده سازی شده در مرحله انتشار رو به جلو^۱ و انتشار رو به عقب^۲ را به صورت مختصر شرح دهید و این دو فرآیند را پیاده سازی کنید.

^۱ Forward Propagation
^۲ Backward Propagation

سوال ۱.۲

یک فایل *.yml برای پارامترها بسازید و پارامترها را با توجه به جدول (۱) مقداردهی اولیه کنید. با استفاده از این پارامترها را شبکه را آموزش دهید. نمودار تغییرات خطا و دقت داده‌های آموزش^۱ و اعتبارسنجی^۲ را در طی دوره‌های مختلف آموزش رسم کنید. همچنین مقدار خطا و دقت را بر روی داده‌های آزمون^۳ بدست آورید.

سوال ۱.۳

تابع هزینه‌ای که در سوال (۱.۲) استفاده کردید، تابع هزینه آنتروپی مقابل^۴ بود. در این سؤال قصد داریم معماری Gaussian-RBF را پیاده‌سازی کنیم. آیا استفاده از تابع هزینه خطای میانگین مربعات^۵ با توجه به ماهیت داده‌ها منطقی به نظر می‌رسد یا نه؟ از این تابع برای آموزش مدل استفاده کنید و نتیجه این قسمت را با سوال (۱.۲) مقایسه کنید.

توضیحات: معماری Gaussian-RBF از تابع هزینه MSE استفاده می‌کند. در این معماری در لایه آخر بایستی تابع Sigmoid بر روی خروجی شبکه اعمال شود.

سوال ۱.۴

در صورتی که وزن‌ها و بایس نوروها را صفر بگذاریم، آیا شبکه آموزش می‌یابد؟ این مسئله را به صورت تئوری و عملی بررسی کنید.

سوال ۱.۵

آیا نرمال / استاندارد کردن داده‌های ورودی تأثیری در دقت مدل دارد؟ این موضوع را به صورت تئوری و عملی نیز بررسی کنید. آیا نتیجه تئوری و عملی بدست آمده مطابق می‌باشند یا خیر؟

سوال ۱.۶

برای آموزش مدل به طور کلی سه روش را در درس زیر فراگرفتیم:

- Batch Gradient Descent
- Stochastic Gradient Descent
- Mini Batch Gradient Descent

¹ Training Dataset

² Validation Dataset

³ Testing Dataset

⁴ Cross-Entropy

⁵ Mean squared error

این سه روش را با یکدیگر مقایسه کنید. برای آموزش مدل با هر کدام از روش‌ها اندازه بسته^۱ را باید چه مقداری بگذاریم؟ تأثیر هر کدام از این سه روش را به صورت عملی بررسی و مقایسه کنید. (برای حالتی که اندازه بسته ابرپارامتر است، حداقل سه اندازه بسته مختلف را مورد بررسی قرار دهید)

در صورت محدودیت‌های زمانی و پردازشی نیازی به بررسی اندازه بسته یک و یا کل دیتاست نمی باشد، اما به صورت تئوری نتایج آن‌ها را پیشبینی کرده و مقایسه کنید.

سوال (۱.۷)

یکی از پارامترهای مهم در آموزش مدل، نرخ یادگیری مورد استفاده در الگوریتم بهینه‌سازی است. این پارامتر را برابر ۰.۰۱، ۰.۱، ۱۰ و ۱۰۰ مقدار اولیه قرار دهید و خطا و دقت مدل را بررسی کنید.

سوال (۱.۸) (امتیازی)

توابع فعال‌ساز ذکر شده در جدول (۱) را با یکدیگر از نظر تئوری مقایسه کنید. همچنین پس از پیاده‌سازی، نتایج آن را مقایسه کنید.

سوال (۱.۹)

تعداد لایه‌ها مدل را از دو لایه به سه لایه افزایش دهید. خطا و دقت بدست آمده بر روی هر کدام از داده‌های آموزش، اعتبارسنجی و آزمون را نسبت به مدل دو لایه (مدل تعریف شده در سوال (۱.۲)) مقایسه کنید.

سوال (۱.۱۰)

دلیل استفاده از تکانه در آموزش شبکه عصبی را شرح دهید. چگونه می‌توان این مفهوم را به صورت عملی پیاده‌سازی کرد و پارامتر آن را تنظیم کنید؟ آیا استفاده از تکانه، باعث بهبود خطا و دقت مدل می‌شود؟

سوال (۱.۱۱)

مشکل بیش‌برازش^۲ در شبکه‌های عصبی را توضیح دهید؟ برای مقابله با این روش چه روش‌هایی وجود دارد؟ حداقل یکی از این روش‌ها را پیاده‌سازی کنید و نتایج را بررسی کنید.

سوال (۱.۱۲) (امتیازی)

دلیل استفاده از K-Fold چیست؟ این روش را به صورت عملی پیاده‌سازی کرده و نتایج را گزارش دهید.

سوال (۱.۱۳) (امتیازی)

در نهایت از نتایج و تمامی پارامترهای ذکر شده در سؤالات پیشین استفاده کرده و بهترین شبکه را آموزش و نتایج را گزارش دهید.

¹ Batch size
² Overfitting

سوال ۲ – مسأله رگرسیون

معرفی داده

داده‌هایی که در این قسمت استفاده می‌کنیم، داده‌های مربوط به قیمت خانه در پاریس است. این داده‌ها دارای ستون‌های زیر است:

- **squareMeter**: متراژ ساختمان
- **numberOfRooms**: تعداد اتاق‌ها
- **hasYard**: وجود یا عدم وجود حیاط
- **hasPool**: استخر داشتن یا نداشتن
- **floors**: تعداد طبقات
- **cityCode: Zip-Code**: مربوط به شهر
- **cityPartRange**: هرچه مقدار این ستون بیشتر باشد، محله منحصر به فردتر است.
- **numPrevOwners**: تعداد مالکین قبلی
- **made**: سال ساخت
- **isNewBuilt**: تازه ساز بودن یا نبودن
- **hasStormProtector**: محافظ طوفان داشتن یا نداشتن
- **basement**: متراژ زیرزمین
- **attic**: متراژ اتاق زیر شیروانی
- **garage**: متراژ گاراژ
- **hasStorageRoom**: انباری داشتن یا نداشتن
- **hasGuestRoom**: تعداد اتاق‌های میهمان
- **price**: قیمت خانه

تعریف مساله

در این سوال قصد داریم با اعمال تغییرات جزئی در ساختار مدلی که در قسمت قبل پیاده سازی کردیم، مدلی را برای دیتاست معرفی شده در بالا تعریف کنیم. در این بخش 70 درصد دادگان را به آموزش و 15 درصد از دادگان را به اعتبارسنجی و 15 درصد باقی مانده را به آزمون اختصاص دهید.

سوال‌ها

سوال ۲.۱)

آیا برای این دیتافریم، پیش پردازش خاصی نیاز است؟ در صورت نیاز، پیش پردازش های لازم را شرح داده و اعمال نمایید

سوال ۲.۲)

از کد بخش اول کمک گرفته و با اعمال تغییرات جزئی در آن کد، مدلی را برای این دیتافریم تعریف کنید. همچنین پارامتر های ذکر شده در قسمت قبل را به نحوی تغییر دهید که به مدلی با عملکرد مناسب (هر چه مقدار تابع هزینه پایین تر، بهتر) دست یابید.

سوال ۲.۳)

نمودار هزینه برای دادگان آموزش، اعتبارسنجی و ارزیابی را رسم کرده و تحلیل نمایید.

نکات تحویل

پیاده‌سازی

- برای پیاده سازی این تمرین فقط مجاز به استفاده از زبان پایتون و کتابخانه NumPy هستید و نمی‌توانید از کتابخانه‌های PyTorch و Tensorflow استفاده کنید.
- کد پیاده‌سازی شده باید با استفاده از کتابخانه NumPy، برداری‌سازی شده باشد. در صورتی که از حلقه استفاده شود، **نمره به شما تعلق نخواهد گرفت**.
- ساختار کد باید به صورت ساختار معرفی شده جدول (۲) باشد.
- از آدرس دهی مطلق در کدهای خود استفاده نکنید و به جای آن از آدرس دهی نسبی استفاده نمایید.
- فایل های ارسال شده باید به فرمت *.py باشد و از ارسال فایل تمرین ها به صورت ipynb خودداری نمائید. همچنین ساختار کلی کدهای شما باید حداقل شامل فایل های زیر باشد.
- همچنین باید کد شما قابلیت اجرا بر روی قسمت کوچکی از داده ها را داشته باشد تا دستیار آموزشی مربوطه بتواند با استفاده از کد شما در مدت زمان کوتاهی مدل شما را آموزش دهد.
- در صورت مشاهده ی موارد تشابه بین دو یا چند فرد در گزارش کار و یا کد ، به طرفین تقلب نمره **صفر** داده خواهد شد. کپی برداری از کدهای آماده موجود در اینترنت و یا استفاده از کدهای افراد ترم های گذشته تفاوت چندانی با تقلب ندارد.
- اگر بخشی از کد را از کدهای آماده اینترنتی استفاده میکنید که جزء قسمت‌های اصلی تمرین نمیباشد، حتما باید لینک آن **در گزارش** و **کد ارجاع** داده شود، در غیر اینصورت تقلب محسوب شده و کل نمره تمرین را از دست میدهید.

گزارش

- گزارش شما در فرایند تصحیح از اهمیت ویژه ای برخوردار خواهد بود. لذا تمامی نکات و فرض هایی که برای پیاده سازی و محاسبات خود در نظر می‌گیرید را در گزارش ذکر کنید. دقت داشته باشید **۵۰ درصد** از نمره تمرین شما مربوط به گزارش است.
- الزامی به ارائه توضیح جزئیات کد در گزارش نمی باشد. اما باید نتایج بدست آمده را به صورت کامل گزارش و تحلیل کنید

- گزارش را به صورت قالب تهیه شده که روی صفحه درس در سامانه ایلرن بارگزاری شده بنویسید. در صورت تمایل می‌توانید از Latex برای نوشتن گزارش استفاده نمایید.
- در گزارش خود برای تصاویر زیرنویس و برای جداول بالانویس اضافه نمایید.

بارگذاری

- به زمان تحویل تمرین ها دقت کافی داشته باشید، برای هر تمرین از زمان ارسال دو هفته فرصت پاسخگویی خواهید داشت.
- مهلت تحویل این تمرین تا پایان روز شنبه ۲۰ اسفند است.
- انجام این تمرین به صورت یک نفره است.
- شما قادر نیستید هیچ تمرینی را با بیش از ۷ روز تاخیر بارگذاری کنید (دقیقا ۷ روز پس از مهلت آپلود، سامانه بسته خواهد شد).
- لطفا برای هر تمرین گزارش ، فایل کدها و سایر ضامم مورد نیاز را با فرمت زیر در صفحه درس در سامانه eLearn بارگذاری نمائید.

HW#_[Lastname]_[StudentNumber].zip

ارتباط با ما

- در صورت وجود هر گونه ابهام یا مشکل می‌توانید از طریق رایانامه زیر با دستیاران آموزشی طراح تمرین در ارتباط باشید:

farshads7778@gmail.com

a.hedayat.m@gmail.com

توضیح	دایرکتوری/فایل
این دایرکتوری شامل مجموعه داده‌های مختلف است که در هر کدام از این زیردایرکتوری‌های مربوط به یک مجموعه داده یک فایل برای پیش‌پردازش داده‌ها وجود دارد.	datasets
این دایرکتوری دارای dataloaderهای مختلف برای مجموعه داده‌های مختلف است.	dataloaders
این دایرکتوری حاوی شبکه‌های عصبی مختلف است.	nets
توابع خطای مورد استفاده و در صورت نیاز مشتق آن‌ها در این دایرکتوری قرار دارند.	losses
در این دایرکتوری کدهای مربوط به training و evaluation است که در آن بر روی مجموعه داده‌ها loop می‌زنیم و فرآیند آموزش یا سنجش مدل را انجام می‌دهیم.	deeplearning
در این دایرکتوری یک سری Utilities مثل توابع مربوط به خواندن فایل *.yaml قرار دارد که در پروژه استفاده می‌شود.	utils
در این فایل شبکه و مجموعه داده نمونه‌گیری می‌شود و تابع train که در deeplearning تعریف کرده‌ایم صدا زده می‌شود.	train.py
در این فایل شبکه و مجموعه داده نمونه‌گیری می‌شود و تابع eval که در deeplearning تعریف کرده‌ایم صدا زده می‌شود.	test.py
پارامترهای آموزش و ارزیابی مدل در این فایل قرار می‌گیرد.	config.yaml