

به نام خدا

دانشگاه تهران

دانشکده مهندسی برق و کامپیوتر



درس شبکه‌های عصبی و یادگیری عمیق

تمرین اول

نام دستیار طراح	حدیثه مصباح	پرسش‌های ۱ و ۲
رایانامه	hadisehmesbah@gmail.com	
نام دستیار طراح	امیرعباس رضا سلطانی	پرسش‌های ۳ و ۴
رایانامه	Amirsoltani2002@gmail.com	
مهلت ارسال پاسخ	۱۴۰۳.۰۸.۱۵	

1	قوانین
1	پرسش 1 - تحلیل و طراحی شبکه‌های عصبی چندلایه (MLP)
1	1-1. طراحی MLP
2	2-1. آموزش دو مدل متفاوت
2	3-1. الگوریتم بازگشت به عقب
2	4-1. بررسی هایپرپارامترهای مختلف
4	پرسش 2 - آموزش و ارزیابی یک شبکه عصبی ساده
4	1-2. آموزش یک شبکه عصبی
5	2-2. آزمون شبکه عصبی بر روی یک دیتاست
7	پرسش 3 - Madaline
7	1-3. الگوریتم‌های MRI و MRII
7	2-3. نمودار پراکندگی داده‌ها
7	3-3. آموزش مدل
7	4-3. تحلیل نتایج
9	پرسش 4 - MLP
9	1-4. نمایش تعداد ستون
9	2-4. ماتریس همبستگی
9	3-4. رسم نمودار
9	4-4. پیش پردازش داده
9	5-4. پیاده سازی مدل
9	6-4. آموزش مدل
10	7-4. تحلیل نتایج

شکل‌ها

شکل ۱. تحلیل نتایج..... 8

قبل از پاسخ دادن به پرسش‌ها، موارد زیر را با دقت مطالعه نمایید:

- از پاسخ‌های خود یک گزارش در قالبی که در صفحه‌ی درس در سامانه‌ی Elearn با نام **REPORTS_TEMPLATE.docx** قرار داده شده تهیه نمایید.
- پیشنهاد می‌شود تمرین‌ها را در قالب گروه‌های دو نفره انجام دهید. (بیش از دو نفر مجاز نیست و تحویل تک نفره نیز نمره‌ی اضافی ندارد) توجه نمایید الزامی در یکسان ماندن اعضای گروه تا انتهای ترم وجود ندارد. (یعنی، می‌توانید تمرین اول را با شخص A و تمرین دوم را با شخص B و ... انجام دهید)
- **کیفیت گزارش شما در فرآیند تصحیح از اهمیت ویژه‌ای برخوردار است؛** بنابراین، لطفا تمامی نکات و فرض‌هایی را که در پیاده‌سازی‌ها و محاسبات خود در نظر می‌گیرید در گزارش ذکر کنید.
- در گزارش خود مطابق با آنچه در قالب نمونه قرار داده شده، برای شکل‌ها زیرنویس و برای جدول‌ها بالانویس در نظر بگیرید.
- الزامی به ارائه توضیح جزئیات کد در گزارش نیست، اما باید نتایج بدست آمده از آن را گزارش و تحلیل کنید.
- **تحلیل نتایج الزامی می‌باشد، حتی اگر در صورت پرسش اشاره‌ای به آن نشده باشد.**
- **دستیاران آموزشی ملزم به اجرا کردن کدهای شما نیستند؛** بنابراین، هرگونه نتیجه و یا تحلیلی که در صورت پرسش از شما خواسته شده را به طور واضح و کامل در گزارش بیاورید. در صورت عدم رعایت این مورد، بدیهی است که از نمره تمرین کسر می‌شود.
- **کدها حتما باید در قالب نوت‌بوک با پسوند ipynb تهیه شوند، در پایان کار، تمامی کد اجرا شود و خروجی هر سلول حتما در این فایل ارسالی شما ذخیره شده باشد.** بنابراین برای مثال اگر خروجی سلولی یک نمودار است که در گزارش آورده‌اید، این نمودار باید هم در گزارش هم در نوت‌بوک کدها وجود داشته باشد.
- **در صورت مشاهده‌ی تقلب امتیاز تمامی افراد شرکت‌کننده در آن، ۱۰۰- لحاظ می‌شود.**
- تنها زبان برنامه نویسی مجاز **Python** است.
- استفاده از کدهای آماده برای تمرین‌ها به هیچ وجه مجاز نیست. در صورتی که دو گروه از یک منبع مشترک استفاده کنند و کدهای مشابه تحویل دهند، تقلب محسوب می‌شود.
- نحوه محاسبه تاخیر به این شکل است: پس از پایان رسیدن مهلت ارسال گزارش، حداکثر تا یک هفته امکان ارسال با تاخیر وجود دارد، پس از این یک هفته نمره آن تکلیف برای شما صفر خواهد شد.

○ سه روز اول: بدون جریمه

○ روز چهارم: ۵ درصد

○ روز پنجم: ۱۰ درصد

○ روز ششم: ۱۵ درصد

○ روز هفتم: ۲۰ درصد

- حداکثر نمره‌ای که برای هر سوال می‌توان اخذ کرد ۱۰۰ بوده و اگر مجموع بارم یک سوال بیشتر از ۱۰۰ باشد، در صورت اخذ نمره بیشتر از ۱۰۰، اعمال نخواهد شد.

○ برای مثال: اگر نمره اخذ شده از سوال ۱ برابر ۱۰۵ و نمره سوال ۲ برابر ۹۵ باشد، نمره نهایی تمرین ۹۷.۵ خواهد بود و نه ۱۰۰.

- لطفا گزارش، کدها و سایر ضمایم را به در یک پوشه با نام زیر قرار داده و آن را فشرده سازید، سپس در سامانه‌ی Elearn بارگذاری نمایید:

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

(مثال: HW۱_Ahmadi_۸۱۰۱۹۹۱۰۱_Bagheri_۸۱۰۱۹۹۱۰۲.zip)

- برای گروه‌های دو نفره، بارگذاری تمرین از جانب یکی از اعضا کافی است ولی پیشنهاد می‌شود هر دو نفر بارگذاری نمایند.

پرسش 1 - تحلیل و طراحی شبکه‌های عصبی چندلایه (MLP)

در دنیای مدرن یادگیری عمیق، طراحی و آموزش شبکه‌های عصبی به یکی از مهم‌ترین چالش‌ها در حوزه‌ی یادگیری ماشین تبدیل شده است. مجموعه داده‌های پیچیده‌ای مانند Fashion-MNIST، که شامل تصاویر دست‌نویس از پوشاک است، به‌عنوان یک استاندارد مرجع برای ارزیابی مدل‌های مختلف شبکه عصبی استفاده می‌شود. در این راستا، پارامترهایی همچون تعداد لایه‌های مخفی، تعداد نودها، استفاده از تکنیک‌هایی مانند Dropout و Regularization، و انتخاب بهترین روش بهینه‌سازی، نقش مهمی در دستیابی به عملکرد بهینه مدل دارند.

در این سوالات، طراحی و آموزش مدل‌های شبکه عصبی با تنظیمات مختلف بررسی شده و تأثیر تنظیمات گوناگون هایپرپارامترها، منظم‌کننده‌ها و روش‌های بهینه‌سازی بر روی مجموعه داده Fashion-MNIST تحلیل می‌شود. در این مسیر، از ماتریس آشفتگی به‌عنوان ابزاری برای بررسی دقیق اشتباهات مدل در تشخیص کلاس‌ها استفاده می‌شود و همچنین روش‌های جستجوی هایپرپارامترها مانند جستجوی بیزی و تصادفی مورد توجه قرار می‌گیرند تا بهترین ترکیب از تنظیمات را انتخاب کنند. این سوالات به شما کمک می‌کنند تا تفاوت‌های کمی و کیفی بین مدل‌های مختلف را درک کرده و از نتایج آن برای بهبود دقت و عملکرد کلی مدل‌ها استفاده کنید. (برای قسمت‌های مختلف سوال که نیاز به مقایسه دارد حتماً از کد برای اثبات گمان‌های خود استفاده کنید.)

۱-۱. طراحی MLP (۲۵ امتیاز)

یک شبکه عصبی چندلایه (MLP) برای مجموعه داده‌ی Fashion-MNIST طراحی کنید که شامل یک لایه مخفی با ۱۰۰ نود و تابع فعال‌سازی ReLU باشد. علاوه بر این، از یک لایه Dropout با نرخ ۳۰٪ و L_2 Regularizer با ضریب $\lambda = 0.0001$ استفاده کنید. این مدل را آموزش دهید و ماتریس آشفتگی را تولید کنید.

- با استفاده از کتابخانه Seaborn یا مشابه، یک Heatmap از ماتریس آشفتگی تولید کنید.
- برای هر کلاس m ، کلاسی را که بیشتر با آن اشتباه گرفته می‌شود، لیست کنید. $m \in \{0, \dots, 9\}$
- دو کلاسی که به طور کلی بیشتر با هم اشتباه گرفته می‌شوند، کدامند؟

- چگونه افزایش پیچیدگی مدل با استفاده از تعداد بیشتر لایه‌های مخفی یا نورون‌ها می‌تواند بهبود عملکرد را در پی داشته باشد؟
- چه معیارهایی برای انتخاب بهترین پیکربندی وجود دارد؟

۲-۱. آموزش دو مدل متفاوت (۲۰ امتیاز)

دو مدل متفاوت با تنظیمات زیر بر روی مجموعه داده Fashion-MNIST آموزش دهید: مدل اول یک لایه مخفی با ۱۲۸ نود و بدون منظم‌کننده و Dropout، مدل دوم یک لایه مخفی با ۴۸ نود و Dropout

با $\text{rate} = 0.2$ و Regularizer L_2 در حالی که $\lambda = 0.0001$ برای هر دو مدل، به مدت ۴۰ دوره

(epochs) آموزش دهید.

- هیستوگرام‌هایی از وزن‌های این دو شبکه تولید کنید – یک هیستوگرام جداگانه برای هر دو لایه (ورودی و مخفی) در هر مدل. تفاوت‌های کیفی بین این هیستوگرام‌ها را توضیح دهید.
- آیا اضافه کردن روش‌های بهینه‌سازی پیشرفته مانند Adam یا RMSprop می‌تواند عملکرد این مدل‌ها را بهبود بخشد؟

۳-۱. الگوریتم بازگشت به عقب (۲۵ امتیاز)

از الگوریتم بازگشت به عقب (Backpropagation) با استفاده از چندین روش بهینه‌سازی (۳ مورد) مانند Adam، Nadam و RMSprop برای آموزش شبکه‌های عصبی بر روی Fashion-MNIST استفاده کنید و آن‌ها را توضیح دهید و مراحل پیاده‌سازی را هم توضیح دهید.

- مقایسه‌ای بین عملکرد این روش‌ها در زمینه سرعت همگرایی و دقت کلی مدل ارائه دهید.
- آیا جستجوی بیزی یا روش‌های دیگر می‌توانند در بهبود این فرآیند تأثیر داشته باشند؟

۴-۱. بررسی هایپرپارامترهای مختلف (۳۰ امتیاز)

هایپرپارامترها را در شبکه عصبی توضیح دهید. اثر تنظیمات هایپرپارامترهای مختلف (۳ مورد) مانند نرخ یادگیری و تعداد نورون‌ها در لایه‌های مخفی بر روی شبکه عصبی برای Fashion-MNIST را بررسی کنید (برای هر کدام حداقل سه حالت متفاوت را امتحان کنید و برای آن نمودارهای Loss و Accuracy

بکشید. در نظر داشته باشید معماری این قسمت ثابت نیست و بر اساس نتایج در قسمت‌های قبلی
میتوانید معماری خود را انتخاب کنید و انتخاب معماری خوب هم شامل نمره می‌باشد).

- توضیح دهید چگونه روش‌های بهینه‌سازی هایپرپارامتر مانند جستجوی تصادفی می‌توانند به انتخاب بهترین ترکیب‌ها کمک کنند؟
- از نتایج ماتریس آشفتگی برای بررسی دقیق‌تر کلاس‌هایی که بیشتر اشتباه گرفته می‌شوند، استفاده کنید و تحلیل کنید تغییر هر کدام از هایپرپارامترها چه تغییری روی کلاس‌هایی که باهم اشتباه گرفته میشوند دارد؟ چرا؟

پرسش ۲ - آموزش و ارزیابی یک شبکه عصبی ساده

۲-۱. آموزش یک شبکه عصبی

در این تمرین، شما کدی خواهید نوشت تا یک شبکه عصبی ساده را آموزش داده و ارزیابی کنید. ما مثالی از فصل ۵ کتاب بیشاپ را دنبال می‌کنیم که از یک لایه پنهان، یک تابع تانژانت هیپربولیک در لایه پنهان، یک identity function در لایه خروجی، و از تابع خطای مربعات (squared error los) استفاده می‌کند. این شبکه شامل ۳۰ نرون در لایه پنهان ($M=30$) و یک نرون در لایه خروجی ($K=1$) خواهد بود. برای پیاده‌سازی این تمرین، از معادلات موجود در فصل ۵ بیشاپ استفاده کنید. (توجه داشته باشید که اجازه استفاده از **Pytorch** یا کتابخانه‌های دیگر را ندارید و فقط می‌توانید از **Python** و **Numpy** استفاده کنید)

الف) یک تابع با نام forward بنویسید که ورودی‌های X ، W_1 و W_2 را گرفته و خروجی‌های z و y_pred را باز می‌گرداند. این تابع فعال‌سازی‌ها را از جلوی شبکه به سمت عقب محاسبه می‌کند. (با استفاده از ویژگی‌های ورودی ثابت و وزن‌ها). همچنین از تابع گذر رو به جلو (forward pass) برای ارزیابی شبکه پس از آموزش استفاده کنید. (۳۰ امتیاز)

• ورودی‌ها:

- یک ماتریس $N \times D$ از ویژگی‌های X ، که N تعداد نمونه‌ها و D تعداد ابعاد ویژگی‌ها است.
- یک ماتریس $M \times D$ از وزن‌ها به نام W_1 بین لایه اول و دوم شبکه، که M تعداد نرون‌های لایه پنهان است.
- یک ماتریس $1 \times M$ از وزن‌ها به نام W_2 بین لایه دوم و سوم شبکه، که یک نرون در لایه خروجی دارد.

• خروجی‌ها:

- یک بردار $N \times 1$ به نام y_pred شامل خروجی‌های لایه آخر برای تمام N نمونه‌ها.
- یک ماتریس $N \times M$ به نام Z شامل فعال‌سازی‌های همه نرون‌های پنهان برای تمام N نمونه‌ها.

ب) یک تابع به نام backward بنویسید که ورودی‌های X ، y ، M ، $iters$ و lr (learning rate) را گرفته و خروجی‌های W_1 ، W_2 و $error_over_time$ را برمی‌گرداند. این تابع آموزش را با استفاده از پس‌انتشار خطا (backpropagation) انجام می‌دهد (و در هر تکرار activation computation function را فراخوانی می‌کند). شبکه را در این تابع بسازید، به این صورت که ماتریس‌های وزن را ایجاد کرده و وزن‌ها را با اعداد تصادفی کوچک مقداردهی اولیه کنید، سپس تکرار کنید ($iters$): یک نمونه آموزشی انتخاب کنید، خطا را در خروجی محاسبه کنید، سپس خطا را به لایه پنهان پس‌انتشار دهید و وزن‌ها را با توجه به خطای به‌دست‌آمده به‌روزرسانی کنید. (۳۰ امتیاز)

• ورودی‌ها:

- یک ماتریس $N \times D$ از ویژگی‌های X ، که N تعداد نمونه‌ها و D تعداد ابعاد ویژگی‌ها (feature dimensions) است
- یک بردار $N \times 1$ از y شامل برچسب‌های درست برای N نمونه
- یک عدد اسکالر M که تعداد نرون‌های لایه پنهان را مشخص می‌کند
- یک عدد صحیح $iters$ که تعداد تکرارها را تعریف می‌کند
- یک عدد اعشاری lr که نرخ یادگیری را تعریف می‌کند
- خروجی‌ها:
- W_1 و W_2 که از قبل برای تابع forward تعریف شده‌اند
- یک بردار $iters \times 1$ به نام $error_over_time$ که خطای نمونه استفاده‌شده در هر تکرار را شامل می‌شود.

۲-۲. آزمون شبکه عصبی بر روی یک مجموعه داده

در تمامی بخش‌های این قسمت نیاز دارید که نمودارهای مختلف را رسم کنید (loss Accuracy) و مقادیر را گزارش کنید. (۴۰ امتیاز)

برای این قسمت شما از مجموعه داده [wine quality](#) استفاده خواهید کرد. هدف این است که امتیاز کیفیت برخی شراب‌ها را بر اساس ویژگی‌های آن‌ها پیدا کنید.

- ابتدا فایل winequality-red.csv را دانلود و بارگذاری کنید. سپس داده‌ها را به دو مجموعه آموزشی و تست (۵۰٪ برای آموزش) تقسیم کنید. داده‌ها را استاندارد کنید، به این صورت که میانگین و انحراف معیار هر بُعد از ویژگی‌ها را تنها با استفاده از مجموعه آموزش محاسبه کرده، سپس از هر ویژگی میانگین را کم کرده و بر انحراف معیار تقسیم کنید (برای هر ویژگی و هر نمونه). به منظور افزودن بایاس به مدل، یک ستون از ۱ها به ابتدای ماتریس ویژگی‌ها اضافه کنید. این کار باعث می‌شود که بایاس به عنوان یک وزن اضافی در مدل لحاظ شود. سپس تعداد نرون‌های لایه پنهان، تعداد تکرارها و نرخ یادگیری را تعیین کنید.
- تابع backward را برای ساخت و آموزش شبکه فراخوانی کنید. از ۱۰۰۰ تکرار (iterations) و ۳۰ نرون پنهان استفاده کنید.
- سپس تابع forward را فراخوانی کنید تا پیش‌بینی‌ها را انجام داده و root mean squared error بین پیش‌بینی‌ها و برچسب‌های واقعی محاسبه شود. عدد آن را گزارش کنید.
- با سه مقدار مختلف برای نرخ یادگیری آزمایش را تکرار کنید. برای هر مورد، خطا را بر حسب زمان (خروجی تابع backward) رسم کنید. این نمودارها را در گزارش خود اضافه کنید. نتایج خود را به طور کامل شرح داده و بهترین حالت را انتخاب کنید. دلیل خود را ذکر کرده و آیا در شبکه‌های عصبی دیگر هم صدق میکند؟ چرا؟

۱-۳. الگوریتم‌های MRI و MRII

در ابتدا یکی از الگوریتم‌های MRI یا MRII را به صورت مختصر توضیح دهید. (۱۰ نمره)

۲-۳. نمودار پراکندگی داده‌ها

با استفاده از کتابخانه pandas در پایتون، ابتدا مجموعه دادگان Question۳.csv را بارگذاری و سپس نمودار پراکندگی داده‌ها را رسم نمایید (ستون اول، ویژگی اول، ستون دوم ویژگی دوم و ستون سوم کلاس داده‌ها را مشخص می‌کند). (۳۰ نمره)

۳-۳. آموزش مدل

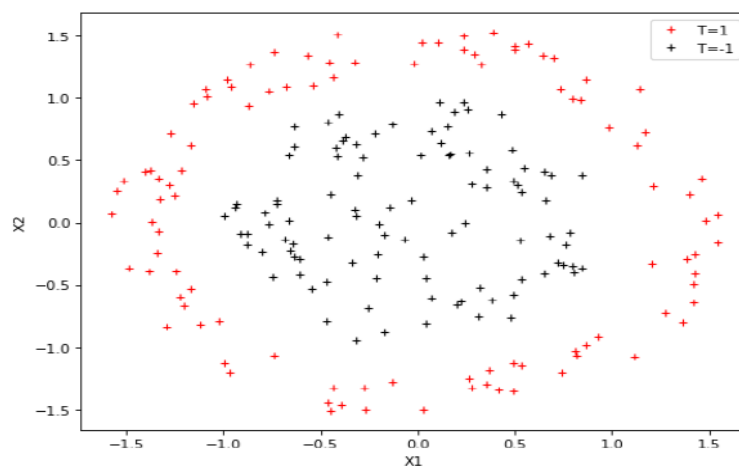
با استفاده از الگوریتمی که در بخش اول مطالعه کردید، شبکه‌ای بر روی داده‌ها آموزش دهید که کلاس داده‌ها را پیش‌بینی کند.

- این داده‌ها را ابتدا با ۳ نورون، سپس با ۴ نورون و در نهایت با ۸ نورون آموزش دهید. (۲۰ نمره)

- خط‌های جداکننده را در هر حالت به صورت مجزا نمایش دهید. (۱۰ نمره)

۴-۳. تحلیل نتایج

در هر حالت مذکور در بخش قبل، دقت و تعداد ایپاک‌ها را با یکدیگر مقایسه و تحلیل کنید. (۳۰ نمره)



شکل ۱. تحلیل نتایج

پرسش ۴ – MLP

۴-۱. نمایش تعداد ستون

یک مجموعه داده حاوی قیمت خانه در اختیار شما قرار گرفته است. فایل Question۴.csv را بخوانید و تعداد Nan های هر ستون را نمایش دهید. (۸ نمره)

۴-۲. ماتریس همبستگی

Correlation matrix را رسم کنید. کدام ویژگی‌ها همبستگی بیشتری با قیمت دارند؟ (۸ نمره)

۴-۳. رسم نمودار

نمودار توزیع قیمت و نمودار قیمت و ویژگی که correlation بیشتری با قیمت دارد را رسم کنید. (۸ نمره)

۴-۴. پیش پردازش داده

ستون date را به دو ستون ماه و سال تبدیل کنید و این ستون را حذف کنید. (۸ نمره)

سپس داده‌ها را به دو بخش train و validation دسته بندی کنید (۲۵ درصد برای validation). (۸ نمره)

با استفاده از MinMaxScaler، داده‌های train و validation را به‌طور جداگانه scale کنید. (دقت کنید که برای scale کردن داده‌های train نباید از داده‌های validation استفاده کنید، زیرا این امر منجر به data leakage خواهد شد). (۸ نمره)

۴-۵. پیاده سازی مدل

در این مرحله دو MLP، یکبار با یک لایه پنهان و یکبار با دو لایه پنهان برای پیش بینی قیمت خانه پیاده کنید. (۸ نمره)

۴-۶. آموزش مدل

optimizer و loss function مناسب را انتخاب کنید. (۸ نمره)

هر کدام از دو مدل را به تعداد epoch مناسب train کنید و نمودار train loss و validation loss در طول یادگیری را نمایش دهید. (۱۴ نمره)

۷-۴. تحلیل نتایج

نتایج نهایی و تعداد epoch مناسب هر دو مدل را با هم مقایسه کنید. علل تفاوت را تحلیل کنید. (۱۴ نمره)

۵ داده از validation set به صورت تصادفی انتخاب کنید و پیش بینی قیمت را با استفاده از مدل بهتر انجام دهید. (۸ نمره)