

بسمه تعالی

پروژه ۲

شبکه عصبی پرسپترون چند لایه (MLP)

وَمَا أَدْرَاكَ مَا شَبْكَه عَصْبِي پرسپترون چند لایه (MLP)!

- مهلت تحویل جمعه، ۱۳۹۹/۰۵/۲۴ ساعت ۲۳:۵۵
 - مهلت ارسال قابل تغییر نیست.
 - مواردی که بعد از تاریخ فوق ارسال شوند نمره ای نخواهند داشت.
 - انجام پروژه تک نفره است. لطفاً به تنهایی انجام شود، در غیر اینصورت نمره منفی در نظر گرفته خواهد شد.
 - کل محتوای ارسالی زیپ شود و نام فایل زیپ ارسالی Project2_studentNumber باشد.
 - زبان برنامه نویسی دلخواه است. (پیشنهاد: پایتون)
 - موارد ارسال شده احتمالاً در تاریخی که بعداً مشخص می‌شود به صورت حضوری/آنلاین نیز تحویل گرفته خواهند شد (صرفاً آنچه در CW طبق تاریخ‌های فوق تحویل داده شده است بعداً به صورت حضوری تست شده و توضیح داده می‌شود)
 - تنها تکالیفی که به CW فرستاده می‌شوند بررسی خواهند شد.
-

شرح:

در این پروژه قصد داریم با شبکه‌های پرسپترون چند لایه و روش آموزش و یادگیری در این شبکه‌ها به صورت عملی آشنا شویم.

در این پروژه استفاده از ابزارها و توابع آماده مربوط به شبکه‌های عصبی (MLP) بلامانع است و توصیه می‌شود از این ابزارهای آماده (مثلاً در پایتون) استفاده کنید.

آنچه شما باید انجام دهید، آماده کردن ورودی و خروجی مورد نظر، طراحی ساختار شبکه، مدیریت فرآیند یادگیری شبکه عصبی و آنالیز نتایج حاصله به منظور اصلاح این فرآیند است.

در هر یک از بخش‌های پروژه که لازم می‌دانستید از تکنیک Cross-Validation استفاده کنید. برخی از قسمت‌های تعریف پروژه نادقیق است و کاری که انجام می‌دهید بستگی به تحلیل‌های خودتان دارد.

برای پروژه یک واسط کاربری گرافیکی نیز طراحی کنید و ورودی و خروجی‌های خواسته شده را در واسط کاربری گرافیکی نمایش دهید. (نیازی نیست به لحاظ ظاهری GUI زیبایی طراحی کنید، فقط کار راه انداز باشد! در هر صورت طراحی آن به عهده شماست).

بخش ۵ و ۶ پروژه نمره و وزن بیشتری دارد.

نوشتن گزارش نهایی این پروژه اهمیت بالایی دارد. ممکن است حتی تحویل حضوری پروژه کنسل شود و صرفاً روی گزارش نهایی شما برای ارزیابی پروژه تاکید شود. بنابراین گزارش جامع و دقیقی از فرآیند اجرای پروژه، آزمایش‌ها، تحلیل‌ها و نتایج حاصله بنویسید.^۱

^۱ - گزارش جوندرا!

بخش اول:

تعدادی تابع (حداقل ۳ تابع و حداکثر بی نهایت) با ورودی یک بعدی از خیلی ساده تا خیلی پیچیده در نظر بگیرید (مثلا یک تابع سینوسی خاص). در یک دامنه مشخص تعدادی نقطه (نمونه) از روی این تابع تولید کنید. (چه تعداد؟). این نقاط را به عنوان مجموعه آموزشی در نظر بگیرید و یا یک MLP سعی کنید خروجی تابع را یادگیرید.

تابعی که شبکه یادگرفته است را در کنار تابع صحیح ترسیم کنید و میزان خطای آنچه یادگرفته شده است را محاسبه (با هر معیاری) و ارزیابی کنید. برای ترسیم تابعی که شبکه یادگرفته، در دامنه ورودی (که ترجیحا وسیع تر از دامنه داده های آموزشی باشد) در بازه های ریز و نزدیک به هم نقاطی را به عنوان ورودی به شبکه بدهید و خروجیش را به دست آورید و نمایش دهید.

با پارامترهای زیر بازی کنید و تاثیر آنها را در نتایج حاصله ببینید و تحلیل کنید:

- تعداد نقاط ورودی
- میزان پیچیدگی تابع مورد نظر
- تعداد لایه های شبکه و تعداد نورون های هر لایه
- تعداد چرخه های شبکه برای تکمیل یادگیری
- وسعت دامنه ورودی خصوصا در توابع پیچیده تر
- هر پارامتر دیگری که فکر می کنید موثر است.

بخش دوم:

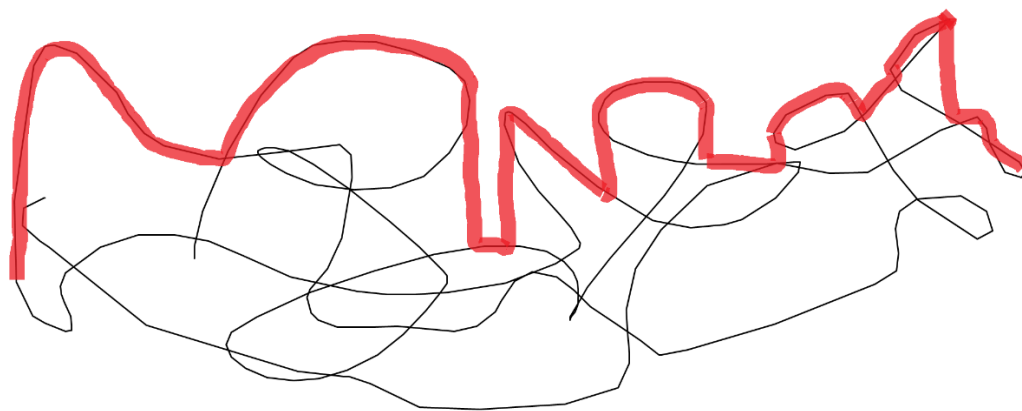
به نقاطی که در بخش قبل تولید کرده اید، نویز (عنصر تصادف) اضافه کنید (مثلا یک مقدار تصادفی کوچک یا بزرگ برای هر نقطه) و تمرین بخش قبل را تکرار کنید. از نویز نزدیک به صفر تا نویز خیلی زیاد را آزمایش کنید. نتایج را با بخش قبل مقایسه کنید و کشفیات و شهودات خود را از این دو آزمایش با پیاز داغ زیاد توصیف کنید.

بخش سوم:

آزمایش بخش اول را برای توابع با ابعاد بالاتر نیز تکرار کنید. برای نمایش توابع با ابعاد بالاتر میتوانید فقط یک بعد (یا نهایتا دو بعد) را نمایش دهید یا ایده دیگری برای نمایش گرافیکی بزنید. حتی می توانید از نمایش گرافیکی صرف نظر کنید و صرفا با معیارهای عددی نتایج حاصله را ارزیابی کنید.

بخش چهارم:

یک قلم و کاغذ آماده کنید(!). هرچه خشم از پروژه‌های درسی و نمره‌ها و اساتید(!) و ... دارید در خود جمع کنید و یک خط فجیع روی کاغذ بکشید و بخش بالایی (یا پایینی) شکل حاصله را به عنوان یک تابع در نظر بگیرید و نقاطی روی این تابع به صورت تقریبی ایجاد کنید (مقدار خروجی تابع را به صورت چشمی تقریب بزنید) و آنها را به عنوان مجموعه آموزشی حاصل از عصبانیت خود به شبکه عصبی بدهید. (مثل شکل‌های زیر)



آزمایش‌هایی که برای یادگیری در این بخش انجام می‌دهید را با آزمایش بخش اول (که تابع مشخصی وجود داشت) مقایسه کنید. سعی کنید با کوچکترین شبکه ممکن (کمترین تعداد نورون) تابعی که ترسیم کرده اید را تقریب بزنید (اندازه این شبکه بیانگر میزان خشم شما خواهد بود! (چرا؟؟)). اگر نمودار ترسیم شده دارای نقاط پرش ناگهانی (مانند شکل‌های بالا) باشد، چه اتفاقی می‌افتد؟

بخش پنجم:

در این بخش به کاربرد دسته بندی توسط شبکه MLP می‌پردازیم (ترجیحا دسته بندی بیش از دو کلاس).

ابتدا لازم است یک پایگاه داده (Data Set) برای خود انتخاب کنید. ترجیحا پایگاه داده‌ای تصویری و یا صوتی انتخاب کنید، اما اگر به کاربرد و زمینه دیگری علاقه‌مند هستید، یا با کمی جستجو علاقه‌مند می‌شوید، می‌توانید در همان زمینه مورد علاقه نیز پایگاه داده مناسب پیدا کنید و روی آن کار کنید. مثلا به عنوان پیشنهاد، می‌توانید پایگاه داده‌های مربوط به شناسایی ارقام دستنویس انگلیسی نظیر USPS یا MNIST (یا اگر برای ارقام دستنویس فارسی پایگاه داده‌ای پیدا کردید) را در نظر بگیرید. یا پایگاه داده‌های تصویر چهره (مانند AR, Yale, Extended Yale B, و ...) و یا هر پایگاه داده دلخواه دیگری که دل در گرو آن بسته‌اید! انتخاب پایگاه داده، سر در آوردن از اینکه آن پایگاه داده چه ویژگی‌هایی دارد، چگونه می‌شود داده‌هایش را Load کرد و چگونه به مجموعه آموزشی و آزمایشی تقسیم شود و جزئی از پروژه و به عهده شما است. یک وقت قابل اعتنا برایش در نظر بگیرید و ساده نگیریدش.

برای پایگاه داده انتخاب شده، با شبکه عصبی، و ترجیحا با استفاده از تکنیک Cross-Validation، اقدام به یادگیری دسته بندی کنید (تعداد داده آموزشی و آزمایشی؟). با پارامترهایی که در بخش اول پروژه گفته شده است، بازی کنید. البته ممکن است زمان اجرای این بخش بالا باشد و نتوان مقادیر زیادی از پارامترها را تست کرد. بنابراین برای داشتن زمان کافی برای اجرای فرآیند آموزش شبکه، بهتر است اجرای پروژه را به هفته آخر موکول نکنید! چون ممکن است زمان آموزش شبکه با تست روی پارامترهای مختلف، حتی یک هفته طول بکشد! ^۲ البته اگر مشکل جدی درباره زمان اجرا داشتید، ناچارید که همه داده‌های آموزشی را در نظر بگیرید و اندازه داده‌های آموزشی را کوچکتر کنید، حتی اگر منجر به کاهش دقت شود.

بخش ششم:

از پایگاه داده انتخاب شده برای بخش پنجم استفاده کنید، ولی این بار برای کاربرد حذف نویز با شبکه عصبی. دسته‌های مختلف را فراموش کنید، مجموعه درهمی از داده‌ها را بردارید، مقداری نویز به آنها اضافه کنید (در تصویر و صوت توابع آماده راحتی برای افزودن نویز می‌توانید پیدا کنید، اگر داده غیر تصویری و غیر نویزی دارید هم اضافه کردن قدری عدد تصادفی به داده‌ها کار دستی ساده‌ای خواهد بود). نسخه با نویز و بدون نویز داده‌ها را

^۲ - در این صورت وقتی در روزهای اجرای برنامه‌تان کسی از شما بپرسد «چه خبر؟» می‌توانید با یک نگاه هوشمند اندر سفیه در وی نظر کنید و بگویید: پروژه ام یه هفته است که زیر باره، دارم run می‌گیرم! (و در دل منظورتان این باشد که متاسفانه CPU های امروزی در حد استعداد شما توانایی و کشش ندارند و اصولا شما به حداقل ۱۰۰ سال آینده متعلق بوده‌اید!)

در نظر بگیرید، نسخه با نویز را به عنوان ورودی و نسخه بدون نویز داده‌ها را به عنوان خروجی شبکه بدهید. آیا شبکه میتواند رفع نویز را یاد بگیرد؟ با چه ساختاری؟ شبکه را در داده‌های آزمایشی تست کنید. رفع نویز در داده آموزشی و آزمایشی چقدر تفاوت دارد؟ این مسیر پرسش و پاسخ را ادامه دهید و تحلیل های خود را بیان کنید. نمونه‌هایی از تصاویر رفع نویز شده را در کنار تصاویر نویزی و تصاویر Original اولیه در کنار هم نمایش دهید تا نتیجه کار شبکه قابل ارزیابی باشد (اگر داده های غیر تصویری انتخاب کرده‌اید هم به نوع دیگری در صورت امکان این کار را انجام دهید). البته در کنار این کار، از معیارهای عددی هم برای ارزیابی دقت حاصله می‌توانید استفاده کنید.

میزان نویز را از خیلی کم تا خیلی زیاد تغییر دهید (حداقل ۳ آزمایش)، و نتایج حاصله را مقایسه و ارزیابی کنید.

بخش هفتم:

مشکل ساختار شبکه در مسائل پیچیده و داده‌های حجیم و بزرگ.

در مسائل پیچیده‌تر، ساختار شبکه به سادگی با آزمون و خطا به دست نمی‌آید. اگر تعداد داده‌های آموزشی زیاد باشد، داده‌ها دارای ابعاد بالایی باشند و آنچه که باید یادگرفته شود (مثلا دسته بندی بین چند کلاس مختلف) دارای پیچیدگی بالایی باشد، آزمودن تعداد لایه‌های مختلف برای شبکه و تعداد نورون‌های مختلف در هر لایه می‌تواند بسیار زمان بر (و در نتیجه غیر عملی) باشد.

آیا از روش‌های تکاملی می‌توان برای حل مشکل فوق کمک گرفت؟ ایده خود را به طور دقیق در چند پاراگراف تشریح کنید.^۳

^۳ - لازم نیست کد بنویسید و پیاده کنید، شمر که نیستیم!

نکته: اگر هفت خوان بالا را به تمام و کمال پشت سر گذاشته اید و از عهده آن برآمده اید، می‌توانید خود را به عنوان یک متخصص هوش مصنوعی به شرکت‌های فنی مختلف معرفی^۴ کنید.

آنچه تحویل داده میشود:

- ۱- کد اجرایی برنامه (دارای واسط کاربری گرافیکی)
- ۲- گزارش کاملی از جزئیات اجرای پروژه که تحلیل‌های خواسته شده در بخشهای مختلف پروژه را در بر بگیرد. لطفاً برای نوشتن گزارش تحلیلی خود، وقت کافی بگذارید. در آینده این گزارش‌ها باید بتواند کارفرمایان را برای اجرای پروژه‌ای که **Prove Of Concept - POC** آن را نوشته‌اید، متقاعد کند.