

مقدمه ای بر یادگیری ماشین

نیمسال اول ۱۳۹۹-۱۴۰۰

مدرس: صابر صالح

تمرین سری چهارم

● مهلت تحویل تمرین های عملی: ۱۳۹۹/۱۰/۰۵ ●

توضیحات

به پیوست تمرین تعدادی فایل ipynb آمده است که هر کدام مربوط به یک بخش از تمرین می باشد. شما می بایست قسمت های مختلف تمرین را در این فایل ها تکمیل کرده و با همین فرمت ارائه شده تحویل دهید. توجه داشته باشید که هر بخش از تمرین که با فرمت ارائه شده متفاوت باشد تصحیح نخواهد شد. در برخی قسمت های تمرین ممکن است که نیاز باشد توضیحات مختصری ارائه کنید. در این موارد، توضیحات را به صورت markdown در نوت بوک هر بخش ذکر کنید. در نهایت همه فایل های تمرین را به صورت یک فایل فشرده به فرمت ML-HW4-9xxxxxxx.zip تحویل دهید.

۱ حل مسئله بهینه سازی در PyTorch

در این بخش می خواهیم با استفاده از ابزار محاسبه گرادیان در pytorch مساله بهینه سازی یافتن نقطه کمینه را برای چند تابع حل کنیم. توجه: در این تمرین مجاز به استفاده از ماژول optim از کتابخانه torch نیستید.

۱.۱ گرادیان کاهشی

در تمرین سری دوم با الگوریتم گرادیان کاهشی که یک الگوریتم بهینه سازی است، آشنا شدیم. در این بخش می خواهیم الگوریتم ساده آن را با یکی دیگر از انواع آن که گرادیان کاهشی با تکانه است، مقایسه کنیم. در الگوریتم گرادیان کاهشی ساده در هر مرحله مقدار تابع را به صورت زیر به روزرسانی می کنیم:

$$\mathbf{x}^{(t+1)} = \mathbf{x}^{(t)} - \eta \nabla L(\mathbf{x}^{(t)})$$

اما در روش گرادیان کاهشی با تکانه، یک پارامتر به صورت تکانه تعریف می کنیم و در هر مرحله مقدار تابع را به صورت زیر به روزرسانی می کنیم:

$$\nu^{(t+1)} = \alpha \nu^{(t)} - \eta \nabla L(\mathbf{x}^{(t)})$$

$$\mathbf{x}^{(t+1)} = \mathbf{x}^{(t)} + \nu^{(t+1)}$$

که در این رابطه ν پارامتر تکانه می باشد و α نیز همانند η یک ضریب ثابت است. حال برای مشاهده اثر این پارامتر به بخش اول فایل Problem1.ipynb مراجعه کرده و موارد خواسته شده را انجام دهید. در انتها نیز با بررسی نتایج به دست آمده اثر این پارامتر را در بهینه سازی به صورت مختصر توضیح دهید.

۲.۱ منظم سازی^۱

منظم سازی یکی از تکنیک هایی است که برای ایجاد تعمیم پذیری بیشتر در مدل از آن استفاده می کنند. برای آشنایی بیشتر با این تکنیک می توانید به این لینک مراجعه کنید.

در این بخش ما با یکی از انواع منظم سازی که برای حل معادلات خطی کاربرد دارد، یعنی ridge regularization آشنا می شویم. در این روش ما با اضافه کردن ضریبی از نرم ۲ وزن های یادگیری شده سعی در آن داریم که وزن ها را به شیوه بهتری یادگیری کنیم. برای مشاهده اثر این روش به ادامه فایل Problem1.ipynb مراجعه کرده و پس از انجام موارد خواسته شده به صورت مختصر اثر بزرگی ضریب regularization را توضیح دهید.

۲ یادگیری مدل ساده در PyTorch

در این بخش برای آشنایی بیشتر با ابزارهای موجود در PyTorch سعی در یادگیری یک مدل ساده چندلایه برای طبقه بندی دیتاست Fashion mnist داریم. در بخش اول تمرین موارد خواسته شده برای یادگیری یک شبکه عصبی ساده را انجام دهید. در بخش بعدی سوال با یکی از روش های Regularization در شبکه های عصبی که dropout نام دارد آشنا می شویم. برای آشنایی بیشتر با این روش می توانید از این لینک کمک بگیرید. موارد خواسته شده در این بخش را نیز انجام دهید و در انتها با استفاده از نتایج به دست آمده و جستجو در اینترنت اثر dropout را به صورت مختصر توضیح دهید. برای توضیحات بیشتر به فایل Problem2.ipynb مراجعه کنید.

¹Regularization

در این بخش می‌خواهیم با استفاده از شبکه‌های عصبی رفع نویز انجام دهیم. دیتاست مورد استفاده Fashion MNIST است. بدین منظور یک دفترچه با نام Problem3 در اختیار شما قرار خواهد گرفت.

۱. ابتدا دیتاست را با استفاده از DataLoader بخوانید. سپس نویزی را با استفاده از Dropout یا توزیع گوسی به دیتاست آموزش اضافه کنید. سپس شبکه اتوانکودر (از نوع تمام متصل) را برای رفع نویز از تصاویر Fashion MNIST تشکیل داده و آموزش دهید. برای آشنایی بیشتر با اتوانکودرها از اینجا کمک بگیرید.

۲. به طور کلی دو نوع اتوانکودر میتوان در نظر گرفت: یکی Undercomplete و دیگری Overcomplete. از هر دو نوع این اتوانکودر استفاده کنید و نتیجه حاصل را بررسی نمایید. برای اطلاعات بیشتر از اینجا کمک بگیرید.

۳. با افزودن جمله منظم‌ساز سعی کنید نتیجه حاصل را بهبود دهید (کدامیک از آنها از جمله L1 یا L2 یا Dropout بر روی عملکرد شبکه بیشتر تاثیر دارد؟). برای آشنایی بیشتر با انواع منظم‌ساز از اینجا کمک بگیرید.