

یادگیری ژرف

تمرین سری پنجم Generative Adversarial Networks

نام استاد: دکتر سلیمانی

نام دانشجو: مهدی یار شهبازی ۹۵۱۰۶۳۹۷

تاریخ تحویل: دوشنبه ۱۳۹۸/۳/۷ دقت شود به خاطر محدودیت در حجم ارسال،برخی از نتبوک ها رو سریع متوقف کردیم بعد از آنکه از عملکرد کامل آنها مطمین شدیم.

سوال ۵

در این سوال تمام مراحل مطابق نت بوک طی شد تنها ذکر ۲ نکته خالی از لطف نیست و آن اینکه به جای آنکه درآپآوت را به شیوه ی خواسته شده بزنیم آن را با تابع F پیاده سازی کردیم که نتیجه ی آن این شد زمانی که مدل را فرامیخوانیم متنی تحت عنوان Dropout نبینم اما در دل مدل استفاده شده است. مورد دوم اینکه به جای MSE از BCE استفاده کردیم که عملکرد بهتری نسیت به MSE داشت.

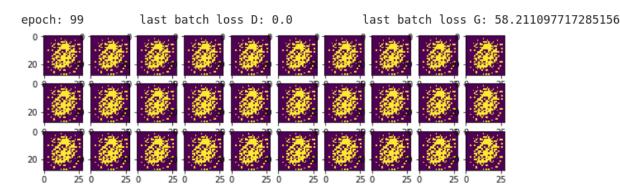
سوال ۶

در این سوال نکته ای که بسیار پر اهمیت بود ترم pullaway loss بود که باعث میشد تنوع نمونه های ما زیاد شود. سایر قسمت مطابق خواسته شده در فایل DL HW5 2 EBGAN.ipynb موجود است.

سوال ٧

الف

در این سوال از ما خواسته شده بود که به در مقداردهی تصادفی، بهجای توزیع نرمال از توزیع یونیفرم استفاده کنیم. در ابتدا از توزیع یونیفرم با ساپورت صفر تا یک استفاده کردیم و نتیجه نهایی قابل مشاهده است. اما زمانی که از یونیفرم با



شكل ١: يونيفرم با ساپورت صفر تا يك

ساپورت منفی یک تا مثبت یک استفاده کردیم، واقعا نتیجه تفاوت معناداری با حالت گوسی مشاهده نشد. و ما این تحلیل را کردیم که حتما باید توزیع مورد استفاده متقارن باشد تا نتایج مطلوب شود. یک نتیجه ی جانبی که این تمرین برای ما داشت، این بود که اگر ما مدل خود را با گوسی ترین کنیم و بعد برای تست از متغیر یونیفرم استفاده کنیم و حتی برعکس اصلا جواب قابل قبولی بدست نمی آید. زیرا مثلا احتمال اینکه یک متغیر تصادفی چند بعدی تمام درایه های آن یک عدد ظاهر شوند، در یونیفرم بسیار بیشتر از گوسی است و خلاصه اینکار امکان پذیر نمیباشد. نتایج یونیفرم با ساپورت متقارن در فایل pytorch(1).ipynb موجود است. دقت شود که خطایی در کد رخ نداده است و از تعمد کد متوقف شده است.

ب

مدل مورد استفاده در این قسمت از یک مدل معروف استفاده برگرفته شدهاست. تنها از .batch norm و به عنوان روش برای پایدار کردن به طور خاص استفاده شد. همچنین تمام توابع فعالساز، leaky relu میباشند. کد این قسمت به نام tensorflow.ipynb ذخیره شده است.