

به نام خدا



دانشکده مهندسی کامپیوتر

درس مبانی هوش محاسباتی

دکتر ناصر مزینی

تمرین سری اول

دستیاران آموزشی:

سپهر باباپور

نیکی نزاکتی

تاریخ تحویل:

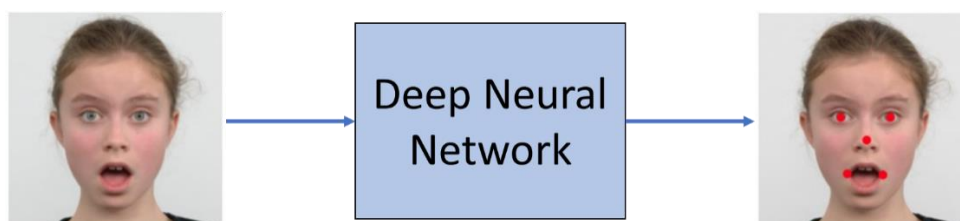
۱۴۰۱/۸/۱۳

نکات تکمیلی:

۱. پاسخ سوالات را به صورت کامل در یک فایل PDF و به همراه فایل کدها در فرمت ipynb. در یک فایل فشرده به شکل HW1_StudentID.zip قرار داده و تا زمان تعیین شده بارگذاری نمایید.
 ۲. برای پیاده سازی ها زبان پایتون پیشنهاد می شود، لازم به ذکر است توضیح کدها و نتایج حاصله باید در فایل PDF آورده شوند و به کد بدون گزارش نمره ای تعلق نخواهد گرفت.
 ۳. در مجموع تمام تمرین ها، تنها ۹۶ ساعت تاخیر در ارسال پاسخ ها مجاز است و پس از آن به ازای هر روز تاخیر ۱۰ درصد از نمره کسر می گردد.
 ۴. چنانچه دانشجویی تمرین را زودتر از موعد ارسال کند و ۷۰ درصد از نمره را کسب کند، تا سقف ۴۸ ساعت به ساعات مجاز تاخیر دانشجو اضافه می گردد.
 ۵. لطفا منابع استفاده شده در حل هر سوال را ذکر کنید.
 ۶. تمرین ها باید به صورت انفرادی انجام شوند و حل گروهی تمرین مجاز نیست.
 ۷. ارزیابی تمرین ها بر اساس صحیح بودن راه حل، گزارش های کامل و دقیق، بهینه بودن کدها و کپی نبودن می باشد.
 ۸. لطفا برای انجام تمرین زمان مناسب اختصاص دهید و انجام آن را به روزهای پایانی موکول نکنید.
 ۹. سوالات خود را می توانید در گروه مربوطه مطرح نمایید (لطفا از پرسیدن سوالات درسی به صورت شخصی خودداری فرمایید، زیرا سوالات بقیه ی دانشجویان هم می تواند مشابه سوالات شما باشد و پرسیدن در فضای عمومی مفیدتر واقع می شود).
- موفق باشید.

۱- در مورد تفاوت **stochastic**، **mini-batch** و **batch** تحقیق کنید. سپس، روش **SGD** را با **GD** مقایسه کنید و مشکلات **SGD** را ذکر کنید و بیان کنید چگونه با استفاده از **Momentum** میتوان این روش را بهبود بخشید. (برای پاسخ به این سوال می‌توانید از [لینک](#) و یا [لینک](#) برای بخش اول و از [لینک](#) برای بخش دوم استفاده کنید).

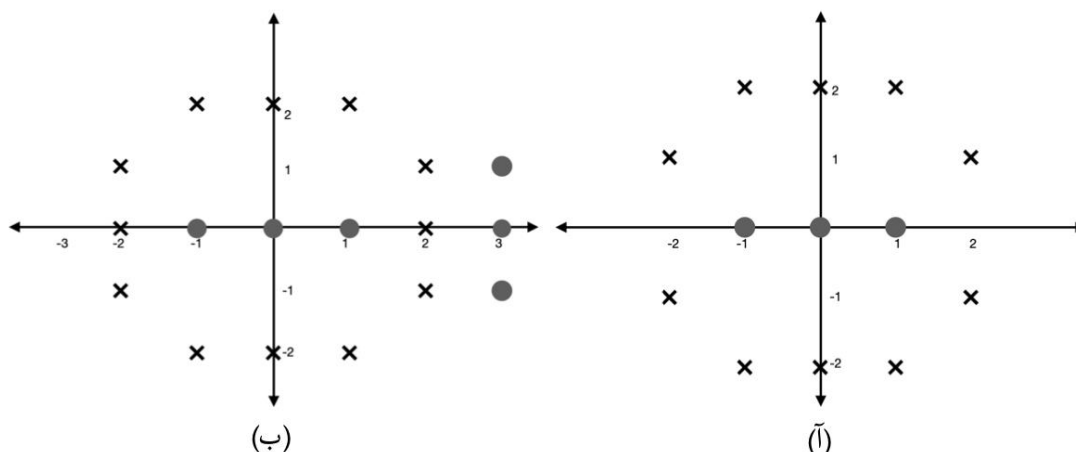
۲- یافتن مکان نقاط مهم در چهره یکی از مراحل بسیار مهم در الگوریتم‌های تحلیل چهره است. همانطور که در شکل زیر نشان داده شده است، ورودی چنین شبکه‌ای یک تصویر برش خورده از چهره است و خروجی آن تخمینی از مختصات نقاط مورد نظر است. در این مثال، خروجی مختصات ۵ نقطه شامل مرکز دو چشم، مرکز بینی و گوشه‌های دهان بوده است که برای نمایش بهتر، بر روی تصاویر اصلی رسم شده‌اند (خروجی شبکه یک تصویر نیست، بلکه خروجی مختصات نقاط است و برای نمایش بهتر بر روی تصویر رسم شده‌اند).



الف) اگر بخواهیم چنین شبکه‌ای طراحی کنیم، در لایه آخر شبکه چند نورون باید داشته باشیم؟ به نظر شما بهتر است از چه تابع فعال‌سازی در لایه آخر استفاده کنیم؟ تابع ضرر مناسب برای حل این مسئله به نظر شما چیست؟ لطفا پاسخ‌های خود را به جزئیات توضیح دهید.

ب) این [کد](#) را بررسی کنید و مشخص کنید در آن برای حل مسئله بالا از چه تابع فعال‌سازی و از چه تابع ضرری استفاده کرده است.

۳- یک پرسپترون تنها می‌تواند داده‌هایی که به صورت خطی قابل جدا کردن هستند را دسته‌بندی کند. توضیح دهید که **Madaline** چگونه مسئله‌ی دسته‌بندی را برای داده‌هایی که غیر خطی هستند حل میکند. آیا می‌توان دیگرام‌های زیر را توسط **Madaline** دسته‌بندی کرد؟ توضیحات لازم را ارائه دهید و در صورت امکان پذیری دسته‌بندی، معماری شبکه عصبی خود را شرح دهید.



۴ - به سوالات زیر پاسخ دهید.

الف) به نظر شما قابلیت تعمیم در کدام یک از شبکه های عصبی **Madaline**، **Perceptron**، **Adaline** و **MLP** بیشتر و در کدامیک کمتر است؟ توضیح دهید.

ب) چه زمانی میگوییم شبکه دچار **Overfit** شده است؟ دلایل مختلف آن را توضیح دهید.

ج) چه روش هایی برای جلوگیری و حل مشکل **Overfit** در شبکه های پرسپترون چند لایه وجود دارد؟

۵ - با استفاده از کتابخانه **NumPy** یک پرسپترون طراحی کنید تا تابع **NOR** را با استفاده از روش

Gradient Descent یاد بگیرد. در نهایت برای اطمینان از درستی آموزش پرسپترون، نموداری شامل خط متمایز کننده ای که توسط پرسپترون آموخته شده است و نقاط نشان دهنده تابع فوق به همراه برجسپ هایشان رسم نمایید.

۶ - با استفاده از کتابخانه **Keras** یک شبکه ی پرسپترون چند لایه طراحی کنید تا عملیات دسته بندی را بر روی

دیتاس **MNIST**، که شامل تصاویر دست نویس اعداد ۰ تا ۹ است، انجام دهد. نمودارهای **دقت** و **خطا** را برای هر **iteration** رسم کنید.

موفق باشید.