

به نام خدا



دانشگاه تهران
پردیس دانشکده‌های فنی
دانشکده برق و کامپیوتر



شبکه های عصبی و یادگیری عمیق

تمرین شماره ۳

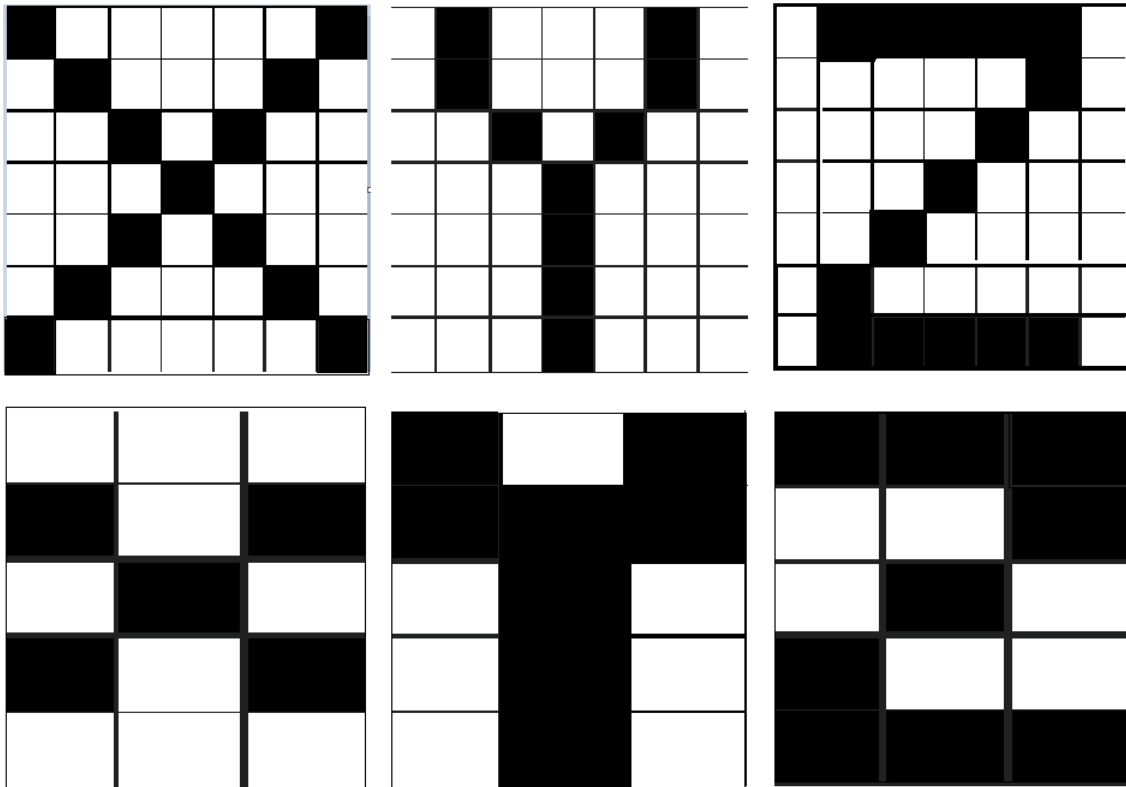
پاییز ۱۳۹۹

فهرست سوالات

- سوال ۱ - *Character Recognition using Hebbian Learning Rule* ۳
- سوال ۲ - *Auto-associative Net* ۴
- سوال ۳ - *Discrete Hopfield Net* ۵
- سوال ۴ - *Bidirectional Associative Memory* ۶

سوال ۱ – Character Recognition using Hebbian Learning Rule

در این سوال می‌خواهیم شبکه‌ای طراحی کنیم که با دادن حروف 7×7 بتواند خروجی متناظر با هر حرف را با ابعاد 5×3 تولید کند. (به خانه‌های سیاه عدد ۱ و به خانه‌های سفید عدد ۰ - اختصاص دهید).



۱) آیا این شبکه می‌تواند تمام ورودی‌ها را به خروجی مطلوب برساند؟

۲) ورودی 7×7 را با نویز ۱۵ و ۴۰ درصد، به صورت تصادفی به شبکه اعمال کنید (به جای ۰، ۱- و به جای ۱، ۰+ قرار دهید). در چند درصد مواقع شبکه قادر به تشخیص درست خروجی خواهد بود؟

۳) ورودی 7×7 را اینبار بگونه‌ای به شبکه اعمال کنید که ۱۵ و ۴۰ درصد از اطلاعات تصویر از بین رفته باشد (به جای مقادیر ۰+ و ۱-، صفر قرار دهید). در چند درصد مواقع شبکه قادر به تشخیص درست خروجی خواهد بود؟

۴) مقاومت شبکه در برابر نویز بیشتر است یا از دست دادن اطلاعات؟ نشان دهید.

۵) مقدار نویز را از ۰ تا ۱۰۰ درصد تغییر داده و نمودار دقت تشخیص را رسم کنید و میزان حداکثر مقاومت شبکه در برابر نویز را بیان کنید.

سوال ۲- Auto-associative Net

در این سوال می خواهیم با استفاده از روش هب به صورت خودانجمنی ماتریس های زیر را ذخیره کنیم:

$$S1 = \begin{bmatrix} 1 & -1 \\ 1 & -1 \end{bmatrix}$$

$$S2 = \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}$$

(۱) وزن های شبکه را با استفاده از Modified Hebbian Learning Rule بیابید.

(۲) تست قوام شبکه را با ۱ و ۲ اشتباه (به جای ۱، -۱، و به جای -۱، ۱+ قرار دهید) به صورت تصادفی در ورودی انجام دهید.

(۳) تست قوام شبکه را در حالتی که ۱ و ۲ تا از درایه های ماتریس ورودی از بین رفته باشد انجام دهید (به جای مقادیر ۱+ و -۱، صفر قرار دهید).

سوال ۳ – Discrete Hopfield Net

در این سوال و با کمک Discrete Hopfield Net می‌خواهیم تصویر سالم (سمت چپ) را به خاطر سپرده و سپس با دادن تصویر خراب (سمت راست) که در آن کلمه‌ی تهران از لوگو خط خورده است، به تصویر سالم برسیم.



برای انجام فرآیند یادگیری به نحوه‌ی زیر عمل کنید.

(۱) تصویر ورودی (شکل سمت چپ) را با استفاده از کد ارسال شده به حالت باینری تبدیل کرده و سائز آن را به ۱۰۰ در ۱۰۰ کاهش دهید.

نکته: در این مرحله نیازی به کد زدن نمی‌باشد و تنها با Run کردن کدی که برای شما قرار داده شده است، به تصویر سیاه و سفیدی که قرار است با آن کار کنید می‌رسید.

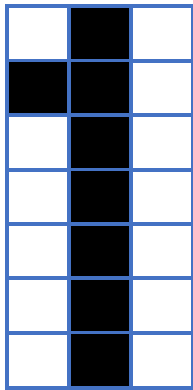
(۲) با استفاده از تصویر باینری ورودی، ماتریس وزن‌ها را بسازید.

(۳) سپس، با کمک ماتریس وزن‌ها و با استفاده از تصویر خراب سعی کنید تا تصویر اصلی و سالم را بازیابی کنید. برای این کار نیاز است که در *iteration* های پیاپی به تصویر سالم نزدیک شده تا اینکه در نهایت به تصویری واضح برسید. *iteration* های خود را برابر با عدد ۳۰۰۰۰ قرار داده و بعد از هر ۱۰۰۰ *iteration* تصویر بدست آمده را رسم کنید تا همگرایی تصویر خراب به تصویر سالم دیده شود. همچنین *theta* یا مقدار آستانه‌ی *activation function* را برابر با ۰.۵ قرار دهید. خروجی شما تقریباً باید شبیه به شکل زیر باشد.

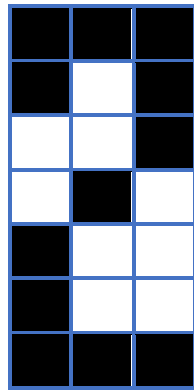
<i>Iteration #1000</i>	<i>Iteration #10000</i>	<i>Iteration #30000</i>

سوال ۴ – Bidirectional Associative Memory

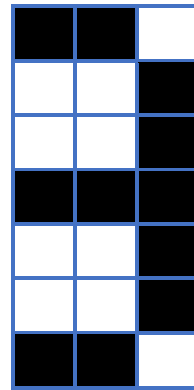
در این سوال از شما خواسته شده است تا شبکه BAM را پیاده‌سازی نمایید. (به خانه‌های سیاه عدد $+1$ و به خانه‌های سفید عدد -1 را اختصاص دهید)



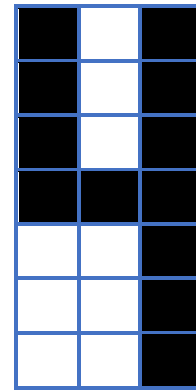
$[-1,1,1,1]$



$[1,-1,1,1]$



$[1,1,-1,1]$



$[1,1,1,-1]$

- (۱) ماتریس وزن مربوط به شبکه BAM را بدست آورید.
- (۲) توانایی شبکه را در بازیابی اطلاعات از ورودی به خروجی و برعکس بررسی کنید. نتایج خود را بصورت کامل گزارش کنید.
- (۳) به هر کدام از ورودی‌ها (اعداد ۱ تا ۴) ۱۵ درصد و ۴۰ درصد نویز اعمال کنید (منظور از اعمال نویز تبدیل خانه‌های $+1$ به -1 و برعکس می‌باشد). برای هر کدام از این دو حالت، کد خود را ۱۰۰۰ بار Run کنید و برای هر کدام از اعداد ۱ تا ۴ درصد دفعاتی که شبکه به خروجی درست رسیده است را گزارش کنید.
- (۴) آیا شبکه با داشتن بردار $[-1,1,1,0]$ به عنوان ورودی می‌تواند خروجی را به درستی شناسایی کند؟ نتیجه خود را گزارش دهید.

نکات:

- مهلت تحویل این تمرین ۱۵ آذر ماه است.
- گزارش را در قالب تهیه شده که روی صفحه درس در Elearn بارگذاری شده، بنویسید.
- گزارش شما در فرآیند تصحیح از اهمیت ویژه‌ای برخوردار است. لطفاً تمامی نکات و فرض‌هایی که برای پیاده‌سازی‌ها و محاسبات خود در نظر می‌گیرید را در گزارش ذکر کنید.
- در گزارش خود برای تصاویر زیرنویس و برای جداول هم بالانویس اضافه کنید.
- الزامی به ارائه توضیح جزئیات کد در گزارش نیست. اما باید نتایج بدست آمده را گزارش و تحلیل کنید.
- دستیاران آموزشی ملزم به اجرا کردن کدهای شما نیستند. بنابراین هرگونه نتیجه و یا تحلیلی که در شرح سوال از شما خواسته شده است را به طور واضح و کامل در گزارش بیاورید. در صورت عدم رعایت این مورد، بدیهی است که از نمره تمرین کسر می‌شود.
- در صورت مشاهده تقلب امتیاز تمامی افراد شرکت‌کننده در آن، ۱۰۰- لحاظ می‌شود.
- برای انجام تمرین‌ها و مینی پروژه‌ها، تنها زبان برنامه نویسی مجاز Python است.
- استفاده از کدهای آماده برای تمرین‌ها به هیچ وجه مجاز نیست. اما برای مینی پروژه‌ها فقط برای قسمت‌هایی از کد و به عنوان راهنمایی برای پیاده‌سازی، می‌توانید از کدهای آماده استفاده کنید.
- نحوه محاسبه تاخیر به این شکل است: مهلت ارسال بدون جریمه تا تاریخ اعلام شده و پس از آن به مدت هفت روز تا ۲۲ آذر بارگذاری ممکن است و در نهایت، پس از بازه تاخیر نمره تکلیف صفر خواهد شد.
- لطفاً گزارش، فایل کدها و سایر ضمایم مورد نیاز را در یک فایل فشرده با فرمت زیر نام‌گذاری و در سامانه مدیریت دروس بارگذاری نمایید.

HW3_[Lastname]_[StudentNumber].zip

- در صورت وجود هرگونه ابهام یا مشکل می‌توانید از طریق رایانامه‌های زیر با دستیاران آموزشی مربوطه در تماس باشید:

○ سوال اول و دوم، مهندس مهسا تاجیک - mahtaj93@gmail.com

○ سوال سوم و چهارم، مهندس سامان ستوده پیم - s.sotoudeh@ut.ac.ir