به نام خدا

دانشکده‌ی علوم و مهندسی کامپیوتر — شبکه‌های عصبی

تکلیف دوم – بهار ۱۳۹٤ — تاریخ تحویل: ۳۱ اردیبهشت ۱۳۹٤

قبل از اینکه شروع به حل تمرین‌ها کنید، حتما یک بار فایل Policies.pdf (موجود در وبگاه) که حاوی نکات مهم در تحویل تکالیف هست را مطالعه فرمایید.
حل سوالاتی که امتیازی هستند، اجباری نبوده و نمره‌ی اضافی خواهند داشت.
طراح هریک از سوالات در زیر مشخص شده است .در صورتی که سوالی دارید می توانید سوالتان را در گروه مطرح کنید یا از طراح سوال بپرسید.

-طراح سوالات ۱ الی ۳: خانم صنیعی
-طراح سوالات ٤ و ٥: آقای کاهانی
-طراح سوالات ٦ و ۷: آقای زندی

سوال ۱:

پژوهش‌های زیادی در حوزه‌ی علوم مغز و بینایی نشان می‌دهند که کارکرد بخشی از مغز که مسئول شناسایی چیزهای مختلف است، مشابه عملکرد شبکه‌های RBF می باشد. ویژگی‌هایی از تصویر که به شناسایی آن کمک می‌کنند و برای دسته بندی تصاویر دریافتی مهم هستند، عموما لبه‌های تصویر در جهات مختلف می باشند. در این تمرین قصد داریم شکل ساده‌ای از طبقه‌بندی با استفاده از شبکه‌های RBF را بررسی کنیم. برای این کار، ۷ کلاس مختلف از مجموعه‌داده‌ی Caltech101 شامل کلاس‌های هواپیما، صندلی، فیل، درخت، موتورسیکلت، توت فرنگی و گل آفتابگردان را انتخاب کرده‌ایم. هر کلاس شامل ۳۰ تصویر است. از ۲۰ تصویر در فاز آموزش و از ۱۰ تصویر دیگر در فاز تست استفاده می‌کنیم.

ورودی شبکه، لبه‌های عمودی، افقی، 45+ و 45- هستند که با استفاده از فیلترهای گابور[1] به دست آمده و به صورت برداری پشت سر هم قرار گرفته‌اند. فایل TrainInputs.mat و TestInputs.mat شامل داده‌های ورودی آماده شده هستند. هر ستون شامل یک نمونه و هر سطر حاوی یک ویژگی نمونه‌های مختلف هستند.

الف) فایل TrainOutputs.mat را بسازید. سپس داده‌های این فایل و فایل TrainInputs را برای آموزش به یک شبکه‌ی RBF بدهید. از دستور newrbe در متلب استفاده کنید. زمان آموزش شبکه را گزارش کنید.

ب) آیا در هنگام آموزش شبکه به خطای خاصی از سوی متلب برمی خوردید؟ اگر پاسخ شما به این سوال مثبت است، چه راه حل هایی برای برطرف کردن این خطا و آموزش موفق شبکه پیشنهاد میکنید؟ (تصاویر در پوشه ی Caltech قرار دارند.)

ج) پس از آموزش شبکه، داده‌های تست را روی آن امتحان کرده و مقدار MSE شبکه را گزارش نمایید.

[1] Gabor Filters

د) سوال بالا را با دستور newrb متلب تکرار کنید. این دستور اجازه تنظیم کردن پارامترهای بیشتری را به کاربر میدهد. با تنظیمات مختلف newrb، بهترین نتیجه از نظر MSE (روی داده های تست) را گزارش نمایید.

سوال ۲:

فایل Tehran2008.xsl حاوی بخشی از اطلاعات آب و هوایی شهر تهران در هر یک از روزهای ۱۲ ماه سال ۲۰۰۸ میلادی است. برای هر روز مشخصات آب و هوایی در چندین ساعت مختلف ثبت شده است. در تمرین سری قبل، دمای هوا را با استفاده از ۵ دمای ثبت شده‌ی قبلی با استفاده از پرسپترون‌های چندلایه پیش‌بینی کردید.

الف) این بار می‌خواهیم شبکه‌ای داشته باشیم که با استفاده از دو ویژگی فشار هوا(P: Pressure) و سرعت باد(ff: Wind Speed) مقدار دما را به عنوان خروجی تولید می‌کند.(یعنی این بار به جای طراحی شبکه‌ی عصبی‌ای که دمای هوا را از روی دمای هوای روزهای دیگر پیش کند، شبکه‌ای طراحی کنید که فشار هوا و سرعت باد در هر روز را به عنوان ورودی می‌گیرد و دمای هوا در آن روز را به عنوان خروجی تولید می‌کند.) دقت کنید که می‌بایست این مساله را با استفاده از تابع newff نرم‌افزار Matlab حل کنید. ساختارهای مختلف شبکه‌ی عصبی شامل 1-5-1، 1-5-5-1، 1-10-1، 1-10-10-1 و 1-20-1 را برای این مساله با مقادیر پیش فرض تابع newrff امتحان کنید.

ب) برای ساختاری که عملکرد بهتری از نظر MSE بر روی داده های تست دارد و با درنظر گرفتن تابع فعالیتpureline برای نورون لایه‌ی خروجی و تابع فعالیت tansig و logsig(هردو را بررسی کنید) برای نورون های لایه ی مخفی، روش‌های یادگیری زیر را یک به یک اعمال کرده و کارایی شبکه را در صورت به کارگیری هرکدام از آن‌ها، با استفاده از MSE مقایسه کنید. زمان آموزش را نیز به ازای به کارگیری هرکدام از روش‌های یادگیری گزارش نمایید.

1)Gradient descent

2)Gradient descent with momentum

3)Gradient descent with adaptive learning rate

4)Levenberg-Marquardt

5)Quasi-Newton

6)Conjugate gradient

-مانند حالت قبل، سطرهایی که اطلاعات مربوط به دما، فشار هوا یا سرعت باد در آن‌ها ناقص است را حذف نمایید.

-تقریبا 2/3 داده‌ها را برای آموزش و 1/3 باقی مانده را برای تست استفاده نمایید.

سوال ۳ (امتیازی):

الگوریتم ژنتیک یکی از الگوریتم‌هایی است که می‌توان از آن برای حل مسائل بهینه سازی استفاده کرد. بنابراین، برای آموزش یک شبکه ی عصبی feedforward نیز میتوان از این الگوریتم بهره برد. در این سوال قصد داریم مساله‌ی تخمین تعداد لکه های موجود در سطح خورشید(Sunspot) در سال‌های آینده، با استفاده از تعداد لکه‌ها در ۴ سال قبل را، با استفاده از یک شبکه‌ی عصبی feedforward که دارای ساختار 4-6-1 است، حل کنیم.

همانطور که می‌دانید، در الگوریتم ژنتیک، هدف این است که یک تابع سازگاری[۲]، کاهش(یا افزایش) یابد. انتخاب کروموزوم های برتر هم بر اساس مقدار تابع سازگاری برای هر کروموزوم در حالت جاری، صورت می گیرد.[۳]

الف) در آموزش یک شبکه‌ی عصبی با استفاده از الگوریتم ژنتیک، کروموزوم‌ها همان وزن ها هستند و تابع سازگاری، MSE است که سعی می‌شود در هر گام کاهش یابد. با استفاده از ابزار ***gatool*** نرم افزار Matlab(که در قسمت ***Optimization*** در نسخه ۲۰۱۳ نرم افزار Matlab موجود است)، یک شبکه ی عصبی را برای حل مساله‌ی Sunspot که در بالا توضیح داده شد، آموزش دهید. برای این منظور از داده‌های فایل sunspot.dat استفاده کنیم. در این فایل، ستون اول نمایشگر سال‌ها و ستون دوم حاوی تعداد Sunspotها در هر سال است. از نمونه‌های ۱۷۰۰ تا ۱۹۳۰ برای آموزش و از نمونه‌های ۱۹۳۱ تا ۲۰۱۲ برای تست استفاده کنید. همانطور که گفته شد، برای پیش‌بینی از ۴ نمونه در زمان‌های قبل از زمان موردنظر استفاده نمایید. هنگامی که تعداد تکرارهای الگوریتم ژنتیک به ۵۰۰ رسید، الگوریتم را متوقف کرده و مدت زمان آموزش را ثبت کنید.

راهنمایی: برای حل این مساله، باید تابعی بنویسید که در هر گام، مقادیر کروموزوم(که در اینجا وزن ها هستند) را به عنوان ورودی می‌گیرد و مقدار تابع سازگاری(که در اینجا MSE شبکه است) را به عنوان خروجی برمی‌گرداند. در قسمت FitnessFunction ابزار ***gatool***، باید نام این تابع را به فرم *@FunctionName* وارد کنید. همچنین در قسمت NumberOfVariables، تعداد متغیرهایی که باید آموزش ببینند را وارد کنید[۴].

---

[2] Fitness Function

[3] در صورتی که با الگوریتم ژنتیک آشنایی ندارید، اطلاعات اولیه و مثال های خوبی را می توانید از بخش‌های مختلف لینک زیر به دست آورید:

http://www.obitko.com/tutorials/genetic-algorithms/index.php

[4] در این زمینه، یا لینک زیر مراجعه کنید:Matlab نرم‌افزار Help برای اطلاعات بیشتر در مورد این ابزار به قسمت

http://www.mathworks.com/help/gads/performing-a-genetic-algorithm-optimization.html

ب) اگر تعداد تکرارها را زیادتر کنید، آیا نتیجه، تغییر چندانی می‌کند؟ شیب تغییر MSE به چه صورت است؟ (نمودار تغییرات MSE در هر تکرار را در گزارشتان بیاورید)

ج) مساله را با همان ورودی‌ها، این بار با الگوریتم آموزش پس انتشار به عقب[5]، حل کنید. زمان آموزش و MSE حاصل از روش ها را با هم مقایسه کنید. کدام یک از روش‌ها از نظر شما بهتر و کارآمدتر است؟ چرا؟

سوال ۴:

در شبکه RBF، با زیاد شدن تعداد نورون های لایه مخفی، اگر فرض کنیم مشکلی برای معکوس کردن ماتریس وجود نداشته باشد. (M: تعداد نورون های لایه ی مخفی – N: تعداد نمونه های آموزشی)

الف) آیا همیشه $M = N$ خطای آزمایش کمتری نسبت به $M < N$ دارد؟ توضیح دهید.

ب) اگر تعداد نمونه ها (N) خیلی زیاد باشد چطور؟

ج) رابطه ی مقدار واریانس کرنل ها (نسبت به حالت بهینه) را با احتمال وقوع Overfitting یا Underfitting بررسی کنید.

سوال ۵:

مجموعه داده ی USPS، متشکل از تصاویر دست نویسی از اعداد ۰ تا ۹ است. که برای سادگی تنها ارقام ۱ تا ۵ استفاده شده است. هر تصویر در سطح خاکستری و به اندازه ی ۱۶ * ۱۶ پیکسل گرفته شده است. تصاویر با یک Row wise به بردار های ۲۵۶ بعدی تبدیل شده و مقادیر پیکسل ها به [-1,1] تغییر مقیاس یافته اند. هدف ایجاد شبکه ای است که فضای ورودی را تغییر داده و ابعاد را کاهش دهد. داده های تست و آموزش یکسان هستند.

فایل Usps5.mat: متشکل از دو ماتریس X و Y است. ماتریس X: نمونه ها به صورت ستونی. ماتریس Y: خروجی مطلوب.

الف) شبکه ای با یک لایه ی مخفی ایجاد کنید که ابعاد ورودی را به ۱۰ بعد کاهش داده و نمونه ها را دسته بندی کند. بهترین دقت بدست آمده را گزارش دهید. (تابع فعالیت های مخفی را خطی در نظر بگیرید)

[5] Backpropagation

راهنمایی: کاهش بعد در لایه ی مخفی انجام گرفته و سپس در لایه ی خروجی نمونه ها دسته بندی می شوند. دقت کنید که ۵ نورون در لایه ی خروجی داریم.

ب) حال اینبار ابعاد را به ۲ کاهش داده و به صورت تصادفی ۱۰۰ نمونه از مجموعه داده را در فضای جدید نمایش دهید (این کارها را برای حالات ۱ و ۲ انجام دهید) و دقت را مقایسه کنید.

۱) توابع فعالیت لایه ی مخفی را خطی در نظر بگیرید.

۲) توابع فعالیت لایه ی مخفی را غیر خطی در نظر بگیرید.

پ) هریک از انواع توابع فعال سازی (خطی – غیر خطی) برای نمونه هایی با چه نوع ویژگی مناسب تر هستند؟

ت) با این شیوه، آیا می توان انتخاب ویژگی (Feature Selection) نیز انجام داد؟ اگر بله، چگونه؟

ث) فرض کنید خروجی مطلوب (Y) همان ورودی (X) باشد، و شبکه ای با M ورودی، N نورون در لایه ی مخفی و M نورون در لایه ی خروجی ایجاد کنیم (N << M). این شبکه چه کاری انجام می دهد؟ تفاوت آن با شبکه ی قبلی به لحاظ کاربردی چیست؟

ج) به طور کلی پس از کاهش بعد، آیا امکان دارد خطا صفر شود؟ (امتیازی)

سوال ۶:

یک نورون را در نظر بگیرید که تابع فعالیت آن f است. تابع f همواره مشتق‌پذیر است. این نورون N ورودی (به جز بایاس) دارد. می‌خواهیم روشی را بررسی کنیم که در آن، فرایند آموزش نیازی به تکرار ندارد. در واقع تنها در یک تکرار جواب مشخص خواهد شد. به علاوه فرض کنیم که قرار است رگرسیون انجام دهیم.

الف- فرض کنید که تابع فعالیت خطی باشد. اگر بخواهیم که خطای آموزش کمینه باشد، فرمول آموزش به صورت زیر خواهد بود. این فرمول را اثبات نمایید.

$$W = (X^T X)^{-1} X^T Y$$

دقت کنید که عملا تکرار نخواهیم داشت و در یک گام وزن‌های نورون طوری بدست می‌آیند که خطای آموزش کمینه است.

X: مجموعه‌ی داده‌ها، هر ردیف یا داده است و ستون‌ها ویژگی‌ها هستند. اندازه: M×F

Y: خروجی مطلوب برای هر یک داده‌ها. M×1

W: وزن‌های مطلوب نورون. F×1

ب- اثبات کنید که خطای آموزش این فرمول کمینه است. (امتیازی)

پ- برای حالتی که تابع فعالیت غیر خطی باشد، فرمول این نوع آموزش چه خواهد شد؟ آیا تابع فعالیت باید شرایط خاصی داشته باشد یا فرمول شما برای هر تابعی صادق است؟

ت- به ازای مجموعه‌ی داده‌ی X، فرمول کمترین خطا چه خواهد شد؟

ث- با توجه به اینکه فرمول کمترین خطای آموزش، تنها در برخی از حالات ورودی‌ها صفر خواهد شد، آیا می‌توان گفت که یک نورون با یک تابع فعالیت خاص، ظرفیت محدودی برای یادگیری دارد؟ به عبارت دیگر اگر بخواهیم کمترین خطای آموزش را داشته باشیم، برخی نورون‌ها هرگز نمی‌توانند این خطا را صفر کنند. پاسخ خود را با ذکر مثال توضیح دهید. (در مثال شما، تابع فعالیت، تعداد ورودی و داده‌ها دلخواه هستند)

ج- توضیح دهید که به طور کلی به نظر شما چه توابع فعالیتی می‌توانند خطای آموزش کمتری داشته باشند؟ (امتیازی)

چ- با وجود چنین روشی که بدون تکرار می‌تواند نورون را آموزش دهد و خطای آموزش آن نیز کمینه است، چه نیازی به روش‌های تکراری است؟ دلایل خود را توضیح دهید.

سوال ۷:

یک نورون را در یک شبکه‌ی پرسپترون چندلایه، در نظر بگیرید. اگر تابع فعالیت tanh باشد:

الف- به صورت ریاضی نشان دهید نورون‌هایی که مقدار v (Induced Local Field) در آنها نزدیک به صفر است، سریعتر آموزش می‌بینند. به عبارت دیگر تغییرات وزن‌های آنها بیشتر است.

ب- برای تابع فعالیت زیر میزان واریانس توزیعی که وزن‌های اولیه‌ی یک نورون از آن انتخاب می‌شوند، باید چقدر باشد؟ (a عدد ثابت است)

$$f(v) = 1.7159 \tanh(av)$$

راهنمایی: از اسلاید ۶۰ الی ۶۲ بخش MLP و همچنین برخی صفحات مقاله‌ی Efficient BackProp استفاده نمایید.