به نام خدا

دانشگاه تهران

پردیس دانشکده‌های فنی

دانشکده برق و کامپیوتر

**سیستم­های هوشمند**

**تمرین کامپیوتری 2**

**فرزاد مهری**

**810194410**

# سوال 1

## تاثیر تعداد نورون های لایه پنهان

با افزایش تعداد نورون های لایه­ پنهان، امکان تطابق با ورودی های غیرخطی­تر و پیچیده تر افزایش می­یابد. به طوریکه در تعداد کم عملا نورون ها نمی­توانند خروجی را به طور مناسب تشخیص دهند. از طرفی با افزایش تعداد نورون­های لایه­ی پنهان، تعداد داده­های مورد نیاز برای آموزش دادن شبکه افزایش می­یابد. با توجه به محدود بودن تعداد داده­های ورودی، اگر تعداد نورون ها بیش از اندازه زیاد انتخاب شود، پس از تعداد زیادی epoch باعث روی دادن over-fitting می­شود. باید تعداد نورون ها به گونه­ای انتخاب شود که بتواند خواص غیر خطی را تشخیص دهد. نمودار همگرایی برای تعداد نورون کم در شکل 1 آمده است. با توجه به شکل می­توان مشاهده کرد که با این تعداد کم نورون، نمیتوان به دقت های بالا دست یافت.



شکل 1، نمودار دقت روی داده های تست به ازای 7 نورون در لایه پنهان



شکل 2، نمودار Loss بر خسب تعداد epoch به ازای 7 نورون

بدون تغییر دادن سایر مشخصات شبکه و فقط با افزایش تعداد نورون ها به 70، نمودار به صورت شکل 2 درمی­آید. که در آن دقت بهتری بدست آمده.



شکل 3، نمودار خطا به ازای 70 نورون در لایه­ی پنهان

## تاثیر سایز هر batch

هر چه سایز batch کوچکتر انتخاب شود، روش gradient descent حساسیت بیشتری به نویز خواهد داشت و احتمال رسیدن به یک local optimum افزایش می­یابد. در این حالت با هر step روش gradient descent ممکن است مقادیر در جهت های مختلف تغییر کند. سرعت همگرایی در این حالت زیاد است. با افزایش سایز batch، سرعت همگرایی کاهش پیدا می­کند اما گرادیان های محاسبه شده در این حالت دقت بهتری دارند و احتمال رسیدن به local optimum کاهش می­یابد. نمودار دقت برای سایز batch 3 و 30 در زیر آمده است. در شکل 4 و 5 اثر نویز و سرعت همگرایی بالا را که ناشی از کم بودن سایز batch است میتوان دید. اما در شکل 6 و 7، نمودار به آرامی و به صورت smooth تغییر کرده و با افزایش سایز batch، واریانس کاهش یافته.



شکل 4، نمودار دقت به ازای سایز batch برابر 3



شکل 5، نمودار loss بر اساس epoch برای سایز batch برابر 3



شکل 6، نمودار دقت برای سایز batch برابر 30



شکل 7، نمودار Loss بر اساس تعداد epoch برای سایز batch برابر 30

## تاثیر تابع فعالساز

به طور کلی انتخاب تابع فعالساز مناسب -که برای تشخیص الگو­های غیرخطی ضروری است- به داده­های ورودی بستگی دارد. از معایب تابع فعالساز tanh و sigmoid این است که گرادیان این توابع در مقدار های زیاد اشباع می­شود و به صفر می­رسد. تابع sigmoid حول صفر متقارن نیست که این باعث سخت شدن بهینه­سازی پارامتر­ها می­شود. همچنین سرعت همگرایی در sigmoid کم است. تابع tanh حول صفر متقارن است و از این رو بهینه­سازی آن آسانتر و همگرایی آن سریع­تر است. tanh به این دلیل بر sigmoid برتری دارد. نمودار دقت برای این توابع فعالساز sigmoid در شکل­های 8 و 9 آمده است. مقایسه این شکل ها با شکل های 6 و 7 که در شرایط مشابه با تابع فعالساز tanh رسم شده اند، تایید کننده برتری tanh از نظر همگرایی و بهینه­سازی بر sigmoid است. همچنین تابع فعالساز f(x)=x برای شبکه­ی عصبی مناسب نیست به علت اینکه یک تابع خطی است.



شکل 8، نمودار دقت با تابع فعالساز sigmoid



شکل 9، نمودار Loss با تابع فعالساز sigmoid

## اثر نرمالیزه­کردن

نرمالیزه­کردن داده­ها با از بین بردن بایاس به سمت feature های ورودی باعث بهبود کارکرد شبکه می­شود و از اشباع شبکه هم جلوگیری می­کند که منجر به افزایش سرعت همگرایی می­شود. روش Z-score و ورش Min-Max هر کدام کاربرد­های متفاوت دارند. روش Min-Max داده­ها را به بازه­ی بین صفر تا یک می­نگارد اما واریانس را کاهش می­دهد. در اینجا تفاوت زیادی بین روش­ها وجود ندارد و فقط Z-score کمی سریع­تر همگرا شده است. نمودار مربوط به حالت نرمالیزاسیون Z-score در شکل 6 و 7 آمده است. شکل 10 و 11 مربوط به نرمالیزاسیون Min-Max است و شکل12 و 13 مربوط به حالت بدون نرمالیزاسیون. با توجه به نمودار ها مشخص است که نرمالیزاسیون بسیار مهم است.



شکل 10، دقت با نرمالیزاسیون Min-Max



شکل 11، Loss با نرمالیزاسیون Min-Max



شکل 12، نمودار دقت بدون نرمالیزاسیون



شکل 13، نمودار Loss بدون نرمالیزاسیون

## تاثیر مقدار­دهی اولیه وزن­ها

مقدار اولیه­ی پارامتر­ها از این نظر اهمیت دارد که مسئله بهینه­سازی شبکه عصبی، یک مساله بهینه­سازی non-convex است و جواب های بهینه­ی محلی متفاوتی ممکن است وجود داشته باشد. با توجه به مقدار اولیه وزن ها، ممکن است حل back-propagation به یکی از این جواب­های بهینه­ی محلی همگرا شود. در شکل 6 و 7، وزن ها به صورت تصادفی در نزدیکی صفر انتخاب شده­اند. در شکل 14 و 15، وزن­ها برابر با مقادیر تصادفی بزرگتری قرار گرفته­اند در نتیجه به یک حل غیر ایده­آل همگرا شده است.



شکل 14، دقت با وزن های با میانگین 0.1



شکل 15، Loss با وزن­های با میانگین 0.1

## تاثیر مقدار نرخ یادگیری

نرخ یادگیری اگر خیلی زیاد انتخاب شود، ممکن است از روی نقطه بهینه جهش کند و منجر به oscillation حول نقطه بهینه شود. همچنین اگر کم انتخاب شود، پیشرفت gradient descent کند می­شود و نمیتوان با داده­های موجود به دقت مورد نظر رسید. در شکل 16 و 17، نرخ یاد­گیری مقدار زیادی در نظر گرفته­شده و در شکل 18 و 19 مقدار کم. نوسان دقت در شکل 16 و 17 به خوبی مشاهده می­شود. همچنین کند بودن همگرایی در شکل 18 و 19 قابل تشخیص است.



شکل 16، نمودار دقت به ازای نرخ یادگیری بزرگ



شکل 17، نمودار Loss به ازای نرخ یادگیری بزرگ



شکل 18، نمودار دقت به ازای نرخ یادگیری کوچک



شکل 19، نمودار Loss به ازای نرخ یادگیری کوچک

## تاثیر coding

کدینگ باینری، با فشرد­سازی بردار خروجی، تعداد نورون­های لایه­ی خروجی را (به ویژه اگر تعداد کلاس های خروجی زیاد باشد) به شدت کاهش می­دهد در نتیجه تعداد وزن­های مورد نیاز کاهش می­یابد. از طرفی با توجه به اینکه در کدینگ باینری، هر بردار خروجی می­تواند بیش از یک مقدار غیر صفر داشته­ باشد، نوعی ارتباط و ترتیب بین داده­های خروجی فرض می­شود. در صورتی که خروجی ها مستقل باشند و ترتیب و ارتباطی بینشان نباشد این مساله باعث کاهش دقت یادگیری شبکه می­شود. در این تمرین نیز به همین دلیل، استفاده از کدینگ one-hot نتایج بهتری میدهد. نمودار دقت برای کدینگ onehot در شکل 20 و 21 و نمودار دقت کدینگ باینری در شکل 22 و 23 آمده است.



شکل 20، نمودار دقت با کدینگ onehot



شکل 21، نمودار Loss با کدینگ onehot



شکل 22، نمودار دقت با کدینگ binary



شکل 23، نمودار Loss با کدینگ باینری

## تاثیر momentum

# سوال 2

بهترین دقت در این مساله به ازای پارامتر­های زیر بدست آمد:

سایز batch: سایز هر batch برابر 30 قرار گرفت تا تعادل بین سرعت همگرایی و دقت جهت گرادیان برقرار شود.

تعداد نورون های لایه پنهان: 70 نورون در لایه­ی پنهان قرار گرفت تا بتواند خواص غیر خطی مساله را تشخیص دهد. افزایش بیشتر تعداد نورون­ها تاثیر زیدادی در دقت نداشت و منجر به افزایش حجم محاسبات میشد.

نرخ یادگیری برابر 0.06 انتخاب شد تا نوسان حول نقطه­ی حل رخ ندهد و جواب با سرعت کافی همگرا شود.

مومنتوم برابر 0 در نظر گرفته شد. با توجه به نرخ یادگیری و سایز batch، در این مساله نیاز زیادی به مومنتم وجود نداشت.

تابع فعالسازی برابر tanh قرار گرفت که بر sigmoid برتری دارد.

داده­های خروجی به صورت one-hot کد شدند تا ترتیبی بین خروجی های مختلف در نظر گرفته نشود و ارتباطی بین آن ها نباشد.

برای پیش­پردازش داده­ها از نرمالیزاسون Z-Score استفاده شد که در این مساله سریع­تر همگرا می­شود.

نمودار بهترین دقت بدست آمده در شکل 24 و 25 رسم شده است. بهترین دقت بدست آمده برابر است.



شکل 24، نمودار بهترین دقت بدست آمده



شکل 25، نمودار Loss در بهترین دقت

# سوال 3

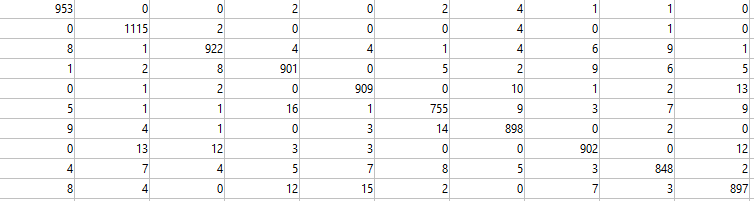
با استفاده از یک کلاسیفایر خطی، دقت طبقه بندی 66% بدست آمد که به مراتب از دقت 91% شبکه عصبی کمتر است. نمودار خطای کلاسیفایر خطی بر حسب epoch در شکل 26 رسم شده است.



شکل 26، نمودار دقت بر حسب تعداد epoch در کلاسیفایر خطی

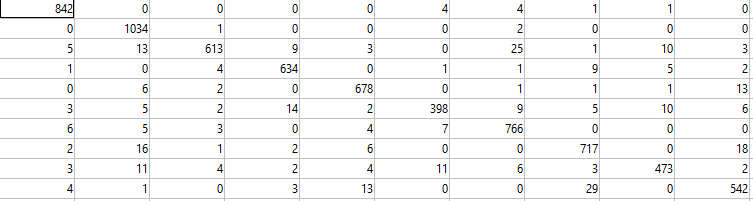
# سوال 4

ماتریس در هم­ریختگی برای طبقه­بندی با استفاده از شبکه عصبی:



شکل 27، ماتریس در­هم­ریختگی، سطر i و ستون j نشان دهنده تعداد داده هایی است که در واقع برابر (i-1) بوده ولی (j-1) تشخیص داده شده است.

ماتریس در­هم­ریختگی برای طبقه­بندی با استفاده از کلاسیفایر خطی:



شکل 28، ماتریس در­هم­ریختگی، سطر i و ستون j نشان دهنده تعداد داده هایی است که در واقع برابر (i-1) بوده ولی (j-1) تشخیص داده شده است.

# سوال 5

فرمول­های محاسبه­ی گرادیان:

1. کلاسیفایر خطی (رگراسیون لاجیستیک)، تابع خطا آنتروپی:
2. شبکه عصبی:

خروجی شبکه:

خروجی لایه­ی پنهان قبل از فعالساز: ، پس از فعالساز:

خروجی لایه­ی خروجی قبل از فعال­ساز:

تابع خطا:

بایاس لایه­ پنهان: ، بایاس لایه­ی خروجی:

تابع فعالساز: f

وزن لایه­ی اول: ، وزن لایه­ی خروجی:

روابط گرادیان به صورت زیر است:

# سوال 6

با استفاده از یک لایه­ی پنهان که تعداد نورون­های آن با توجه به میزان فشرده­سازی مورد نطر انتخاب می­شود، میتوان ابعاد داده را کاهش داد. در اینجا به ازای سه حالت مختلف 64، 128 و 256 نورون در لایه­ی پنهان ابعاد داده کاهش داده شده است. بدیهی است به طور کلی هر چه از تعداد نورون­های بیشتری استفاده شود، می­توان با دقت بهتری داده­ها را بازیابی کرد. تعداد نورون­ها باید به گونه­ای انتخاب شود که با کمترین تعداد ممکن به دقت مورد نظر دست یافت. در اموزش شبکه از نرمالیزاسیون max-min استفاده شده است. نمودار Loss بر حسب تعداد epoch در شکل­های 29، 30 و 31 رسم شده است.



شکل 29، نمودار Loss بر حسب تعداد epoch برای حالت 256 نورون در لایه­ی پنهان



شکل 30، نمودار Loss بر حسب تعداد epoch برای حالت 64 نورون



شکل 31، نمودار Loss برای 128 نورون

با توجه به این که در 256 نورون، دقت بالاتر است از آن استفاه شده است. همچنین برای اینکه فشرده­سازی قابل ملاحظه­ای وجود داشته باشد، از نورون های بیش­تر استفاده نشده است.

از خروجی 256 بعدی لایه­ی پنهان، برای اموزش شبکه استفاده شده است. نمودار دقت و Loss برای این حالت در شکل 32 و 33 رسم شده است. با توجه به نمودار­ها مشخص است که دقت شبکه به مقدار قابل توجهی کاهش یافته است از آنجایی که از feature های تقریبی استفاده شده است، بخشی از اطلاعات موجوددر ورودی از بین رفته است و مقداری نویز در نتیجه­ی این تغییر به داده­های ورودی اضافه شده است که منجر به کاهش دقت شده است.



شکل 32، نمودار دقت با feature های فشرده شده



شکل 33، نمودار Loss با feature های فشرده شده