

به نام خدا

عنوان:

تکلیف دوم درس شبکه های عصبی-RBF

استاد:

دکتر منصوری

نویسنده:

محمد علی مجتهد سلیمانی

۴۰۳۳۹۰۴۵۰۴

تاریخ:

۱۴۰۳/۰۸/۲۸

◆ سوال ۱.

شبکه های RBF یک نوعی از شبکه های عصبی مصنوعی هستند که از radial basis به عنوان تابع فعالیت استفاده میکنند. ساختار آنها به این شکل است که لایه ورودی دارند. فقط یک لایه مخفی دارند معمولاً از تابع gaussian برای این لایه استفاده میشود تا ورودی را تبدیل کند. واکنش نود های میانی به فاصله بین ورودی و نقاط مرکز بستگی دارد. لایه خروجی، خروجی شبکه را تولید میکند که یک ترکیب خطی از خروجی لایه میانی است.

• کاربرد RBFها:

✓ شناسایی الگو یا کلاس بندی:

میتوانند برای شناسایی الگو ها در داده هایی مثل عکس ها یا تشخیص گفتار یا تشخیص چهره مورد استفاده قرار بگیرند.

✓ تابع تخمین:

این شبکه ها به صورت گسترده میتوانند برای تخمین زدن های ریاضی به خصوص وقتی ارتباط میان ورودی و خروجی غیر خطی و پیچیده است مورد استفاده قرار بگیرند. به سبب اینکه نسبت به داده های نویز تحمل پذیری بیشتری دارند. و به صورت طبیعی میتوانند خوشه ها را مدیریت بکنند از طریق تابع های شعاعی و همچنین یادگیری سریعتری نسبت به backpropagation دارند.

✓ پیشبینی سری زمانی:

برای پیشبینی داده های سری زمانی مثل بازار های بورسی مورد استفاده هستند. بخاطر اینکه میتوانند نسبت به وابستگی های موقتی بهتر عمل کنند. نسبت به یادگیری الگو های تکرار شونده کارآمد هستند. کمتر مستعد بیش بردازش هستند.

چرا از شبکه های عصبی سستی بهتر هستند؟ زمان کمتری برای یادگیری نیاز دارند. تعمیم پذیری بیشتری دارند. تحمل پذیری بیشتری نسبت به داده های نویز دار دارند و در نهایت پیاده سازی راحت تری دارند.

◆ سوال ۲.

در ۳ سناریو میتوان تعداد نورون ها را تحلیل کرد ۱. تعداد نورون ها کم باشد. ۲. تعداد نورون ها مناسب باشد. ۳. تعداد نورون ها زیاد باشد.

۱. اگر تعداد نورون ها کم باشد مدل با وجود اینکه پیچیدگی کمی دارد، اما ظرفیت محدودی برای ذخیره و مدل کردن الگو های پیچیده دارد. هر نورون میتواند ناحیه کوچکی از فضا را پوشش دهد نسبت به ورودی هایی که در آن فضا وجود دارند. مدل شاید بتواند الگو های خطی یا الگو های ساده تر را ذخیره کند اما نسبت به الگو های پیچیده مدل ناتوان است به سبب اینکه پوشش ناکافی دارد نسبت به فضا.

۲. اگر تعداد نورون ها مناسب باشد، شبکه ظرفیت کافی برای پوشش نواحی مختلف و مهم فضا را دارد، و مدل میتواند بسیار منعطف باشد نسبت به الگو های ساده و پیچیده و این قاعده باعث

میشود که ما بتوانیم یک **trade off**ی بین پیچیدگی شبکه و تعمیم پذیری آن برقرار کنیم و هر ۲ متوازن باشند.

۳. اگر تعداد نورون ها خیلی زیاد باشد، شبکه بسیار پیچیده میشود و شبکه مستعد **overfit** شدن میشود به سبب اینکه خیلی خوب میتواند نسبت به داده آموزشی، سازگار شود اما نسبت به داده های جدید بطور ضعیفی عمل کند. محاسبات برای نورون های زیاد پیچیده و زیاد میشود.

♦ سوال ۳.

نحوه آموزش شبکه های **RBF** با تکیه بر پارامتر های گفته شده دارای یک سری گام هست: مشخص کردن مراکز و عرض شبکه و بهینه سازی وزن های خروجی.

✓ برای مشخص کردن مراکز در فضای ورودی روش های متعددی وجود دارد: تکنیک های خوشه بندی، انتخاب تصادفی و رویکرد یادگیری نظارت شده.

- تکنیک های خوشه بندی: به عنوان مثال در این تکنیک الگوریتم **K-MEAN** مورد استفاده قرار میگیرد تا مراکز را مشخص کند. ورودی ها دسته بندی میشوند به یک سری از خوشه ها و مراکز این خوشه ها میشوند مراکز شبکه ما.

- انتخاب تصادفی: در این رویکرد یک زیر مجموعه از نقاط را به صورت تصادفی به عنوان مراکز انتخاب میکنیم.

- یادگیری نظارت شده: در این رویکرد بر اساس موارد بهینه سازی سعی میکنم به صورت چرخشی مراکز را بدست بیاوریم.

✓ برای پیدا کردن عرض مناسب یا همان **width** رویکرد های مختلفی داریم: عرض ثابت، روش های وفقی و تکنیک های بهینه سازی.

- عرض ثابت: در این روش یک مقدار ثابت را که معمولاً بر اساس فاصله بین داده ها و مراکز هست را به عنوان عرض در نظر میگیریم.
 - روش های افقی: در این روش عرض ها به صورت پویا تغییر میکنند و تنظیم میشوند بر اساس فاصله همسایگی هر مرکز.
 - تکنیک های بهینه سازی: میتوانیم از روش های جستجو گرادیان یا **grid** برای پیاده کردن عرض مناسب در طی فرآیند یادگیری استفاده بکنیم.
 - ✓ برای پیدا کردن وزن های خروجی، بعد از اینکه مراکز و عرض را مشخص کردیم شبکه **RBF** به عنوان یک سیستم خطی عمل میکند که خروجی جمع وزن دارد فعالیت های **RBF** هستند. وزن های خروجی بهینه سازی شده اند تا میزان خطا بین خروجی واقعی و هدف را کاهش دهند و از یک سری روش ها استفاده میکنند مانند: **least square** و گرادیان کاهشی.
 - **Least square**: که همان مسئله **regression** خطی ما هست برای پیدا کردن وزن هایی که این تابع را کاهش دهند.
 - گرادیان کاهش: با کمک الگوریتم **backpropagation** سعی میکنند وزن های مناسب را پیدا کنند. این خصوصیت اجازه یادگیری **online** را میدهد.
- با بهینه سازی پارامتر گفته شده شبکه **RBF** اقدام به یادگیری و تخمین زدن ورودی های جدید میکند و در ادامه با تمرکز بر الگوریتم **K-MEAN** جلو میرویم. خوشه بندی **K-MEAN** به صورت گسترده به عنوان یک روش مهم برای انتخاب اولیه مراکز استفاده میشود. این روش داده ها به خوشه های **K** تایی انتخاب میکند و مراکز این خوشه ها به عنوان مراکز شبکه استفاده میشوند. الگوریتم سعی دارد واریانس را کاهش دهند تا نشان دهد مراکز واقعاً نسبت به توزیع ورودی، پوشش دهنده هستند.

• انتخاب روش **K-MEAN** برای مراکز:

- همگرایی: همگرایی در این روش سریعتر میشود. انتخاب خوب عرض ها باعث میشود که پوشش خوبی از فضای ورودی ایجاد شود. بهینه سازی خطی وزن های خروجی هم مناسب است.
- بیش بردازش: در پدیده بیش بردازش انتخاب مراکز به عنوان پیچیدگی عمل میکنند، اگر مراکز بتوانند خوب توزیع شوند بر اساس دسته های ورودی خطر بیش بردازش کاهش پیدا میکند. انتخاب عرض مناسب باعث میشود تا تعمیم پذیری خوب باشد.
- مقاومت در برابر نویز: در برابر نویز هم انتخاب عرض های بزرگتر باعث میشود در بردار نویز تاب آوری داشته باشد البته از طرف دیگر دقت کاهش پیدا میکند. اگر داده ها دارای outlier باشند مستعد نویز هستند و باعث میشود مراکز خوشه ها منحرف شوند، بهتر است قبل از استفاده از این روش یک پیش پردازشی صورت بگیرد تا داده های outlier حذف شوند.

• انتخاب روش گرادیان کاهش برای وزن های خروجی:

- همگرایی: برای وزن های خطی به خصوص وقتی مراکز و عرض ها بهینه شده باشند همگرایی سریع خواهد بود.
- بیش بردازش: اگر regularization انجام شود خطر کاهش پیدا میکند، زیرا وزن های بیش از حد بزرگ را که باعث سازگار شدن نویز روی داده ها میشود را منع میکند.
- مقاومت در برابر نویز: مقاومت بالاتر می رود زیرا بخاطر regularization باعث میشود که نسبت به داده های آموزش نویز دار بیش از حد وفق پیدا نکند.

• استفاده از روش های وفقی برای عرض:

استفاده از مقدار ثابت برای عرض ممکن است نتوانیم به خوبی تخمین بزنیم.

- همگرایی: روش های افقی، همگرایی را کند میکنند بخاطر اینکه پیچیدگی بیشتر میشود. البته این افزایش پیچیدگی باعث میشود بازدهی بهتری از لحاظ عملکرد نسبت به توزیع های مختلف داده داشته باشند.
- بیش برداش: روش های افقی کمک میکند تا از بیش برداش در نواحی مختلف جلوگیری شود.
- مقاومت در برابر نویز: بر خلاف مقدار ثابت برای عرض، روش های افقی به سبب اینکه به صورت محلی وفق پیدا نمیکنند با یک مقدار ثابت تاب آوری بیشتری در برابر نویز دارند.

*سوال ۴ در صفحه بعد

$$\sigma = 0,1 \quad | \quad C_1 = 2, C_2 = 5, C_3 = 1$$

$$\phi_i(x) = e^{-\frac{(x-c_i)^2}{2\sigma^2}}$$

وزن لایه = برای لایه مخفی :

$$\omega_1 = 1,5, \omega_2 = -0,1, \omega_3 = 0,2$$

$$\text{Bias: } b = -0,2$$

$$\phi_i(4) = e^{-\frac{(4-c_i)^2}{2\sigma^2}}$$

ورودی $x = 4$

جمع وزن دار، مخفی

$$y = \omega_1 \phi_1(x) + \omega_2 \phi_2(x) + \omega_3 \phi_3(x) + b$$

$$C_1 = \phi_1(x) = e^{-\frac{(x-2)^2}{(2 \times 0,1)^2}} = e^{-\frac{4}{1,2}} = e^{-3,125} = 0,05431$$

$$C_2 = \phi_2(x) = e^{-\frac{(x-5)^2}{(2 \times 0,1)^2}} = e^{-\frac{1}{1,2}} = e^{-0,1711} = 0,14571$$

$$C_3 = \phi_3(x) = e^{-\frac{(x-1)^2}{(2 \times 0,1)^2}} = e^{-\frac{14}{1,2}} = e^{-11,67} = 0,0000027$$

$$1,5(0,05431) + (-0,1)(0,14571) + 0,2(0,0000027) + (-0,2) = y \approx$$

$$y = -0,15005 \rightarrow \text{for } \sigma = 0,1$$

$$y \text{ for } \sigma = 0,1 \quad \phi_1(x) = 0,05431 \times 10^{-4}$$

$$\phi_2(x) = 0,14571$$

$$\phi_3(x) = 1,93 \times 10^{-22}$$

$$y = -0,2321$$

همانطور که مشاهده میشود وقتی سیگما را کاهش میدهیم و نصف میکنیم تابع gaussian ما باریک تر میشود و قله های ما تیز تر میشوند. جواب ها در اطراف مراکز بیشتر محلی میشوند. و فقط ورودی های نزدیک به مرکز تاثیر گذار هستند و شبکه سخت تر میشود و بسیار حساس میشود نسبت به تغییرات محلی. در نتیجه تعمیم پذیری مدل آسیب میبیند. نیاز داریم به افزایش تعداد مراکز برای پوشش تمام فضای ورودی، و خروجی برای ورودی هایی که دور از مراکز هستند نزدیک به صفر میشود.