به نام خدا

یادگیری ماشین

تمرین چهارم

ایمان رسولی پرتو

۸۱۰۱۹۹۴۲۵

۱.

الف) هدف توابع فعال‌ساز در شبکه‌های MLP غیرخطی کردن شبکه، افزودن قابلیت backpropagation و در نتیجه ایجاد قابلیت چندلایه‌سازی شبکه است. با چندلایه شدن و عمیق شدن شبکه امکان یادگیری توابع پیچیده‌تر با پارامترهای بیشتر فراهم می‌شود.

اگر از توابع فعال‌ساز استفاده نشود، خروجی شبکه خطی خواهد بود و هر چقدر لایه‌ها افزایش پیدا کنند قابل ادغام با لایه‌های قبلی هستند و در نهایت خروجی همچنان ترکیب خطی از ورودی‌هاست. همچنین توانایی یادگیری توابع پیچیده را دیگر ندارد و عملیات backpropagation نیز دچار مشکل می‌شود.

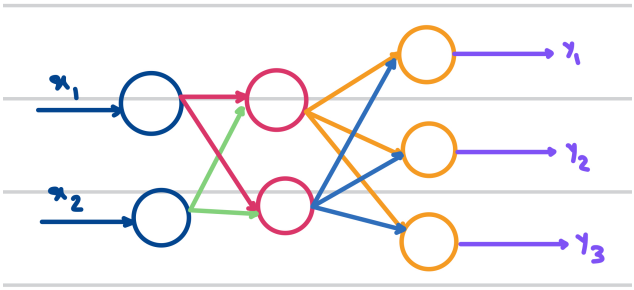
ب) اگر ورودی تابع sigmoid بسیار بزرگ یا کوچک باشد، تغییرات خروجی کم است به این مغنی که گرادیان نزدیک به صفر می‌شود. این داستان باعث کندشدن سرعت همگرایی می‌شود.

محدود بودن خروجی تابع sigmoid بین ۰ و ۱ و مرکزیت صفر نداشتن باعث آپدیت‌های غیربهینه در gradient descent می‌شود که باعث کندشدن سرعت همگرایی می‌شود.

یکی از جایگزین‌ها تابع فعال‌ساز ReLU می‌باشد. این تابع مشکل ورودی کوچک و بزرگ را برطرف می‌کند.همچنین پیاده‌سازی ساده‌تری نسبت به sigmoid دارد و بهینه‌تر است.

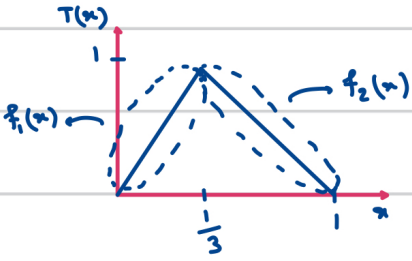
جایگزین مناسب دیگر tanh است که مرکزیت صفر دارد و آپدیت‌های گرادیان را بهینه‌تر انجام می‌دهد.

ج) چون دو ویژگی داریم شبکه دو ورودی خواهد داشت. چون سه کلاس داریم شبکه سه کلاس دارد. داده‌ها با دو خط از هم جدا می‌شوند و یک لایه میانی کافی است. در لایه خروجی تابع فعال‌ساز softmax قرار می‌دهیم. در نتیجه شبکه به شکل زیر توصیف می‌شود:



شکل ۱: ساختار شبکه طراحی شده

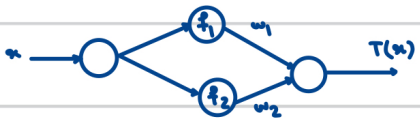
د)



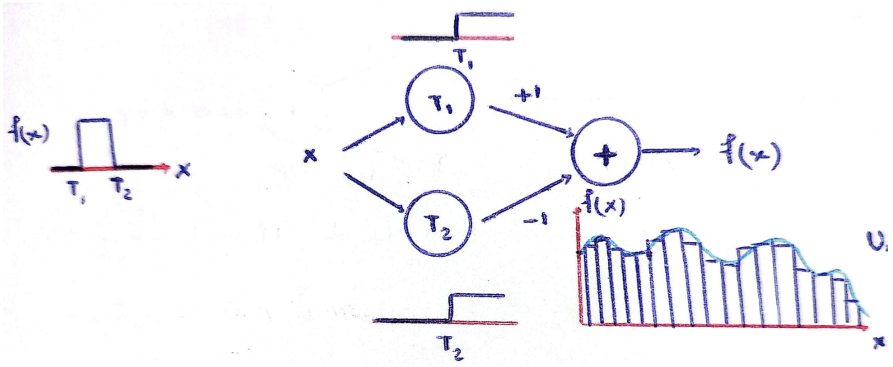
$$T(x) = f_1(x) + f_2(x) = \underbrace{3\,\mathrm{ReLu}(x)}_{w_1} - \underbrace{\frac{3}{2}\,\mathrm{ReLu}(x-1)}_{w_2}$$

$$f_1(x) = 3\,\mathrm{ReLu}(x)$$

$$f_2(x) = -\frac{3}{2}\,\mathrm{ReLu}(x-1)$$

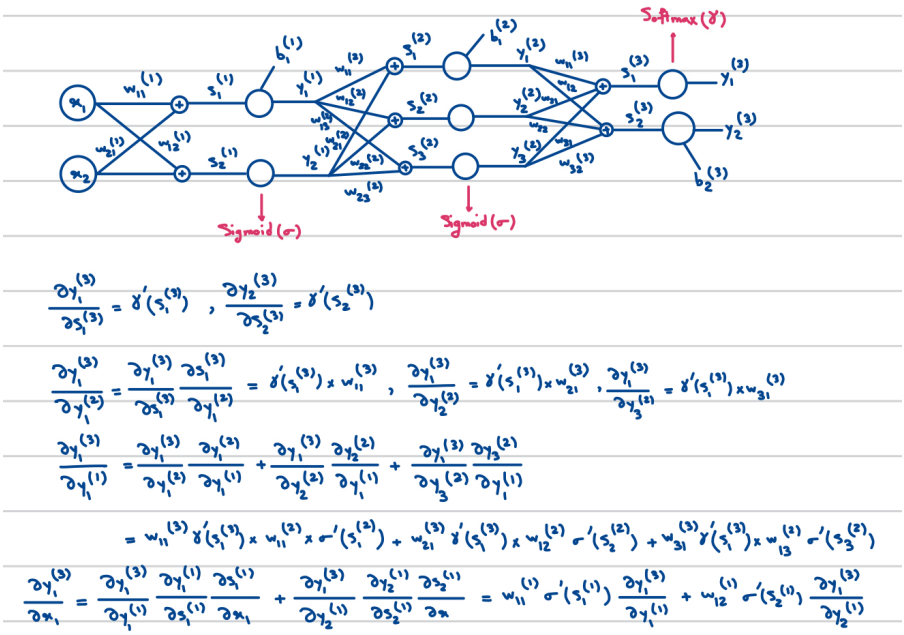


ه) با استفاده از دو لایه می‌توان تابع پالس ایجاد کرد. هر تابع دلخواهی را می‌توان به شکل ترکیب خطی چند پالس بیان کرد. در نتیجه این مدل MLP می‌تواند یک Universal appropriator برای هر تابعی باشد.



۲.

الف)



ب)

Stochastic gradient descent : مشابه الگوریتم Gradient descent می‌باشد. در هر تکرار بجای استفاده از کل داده‌ها برای محاسبه گرادیان تنها از یک نمونه یا یک زیرمجموعه از کل داده‌ها استفاده می‌شود. اینکار باعث کاهش حجم محاسبات و افزایش سرعت الگوریتم می‌شود.
نحوه کار: الگوریتم با مقداردهی اولیه به وزن‌ها آغاز می‌شود سپس یک داده یا زیرمجموعه‌ای از کل داده‌ها به شکل تصادفی انتخاب می‌شود. براساس داده(داده‌های) انتخاب شده، گرادیان محاسبه شده و وزن‌ها آپدیت می‌شود. این مراحل (غیر از مقداردهی اولیه) تا زمان همگرایی تکرار خواهد شد.

Newton–Raphson method : این الگوریتم یک روش تکراری برای محاسبات ریشه توابع حقیقی است.
نحوه کار: ابتدا یک نقطه به شکل تصادفی به عنوان نقطه شروع انتخاب می‌شود. سپس مشتق تابع در آن نقطه محاسبه می‌شود و نقطه طبق رابطه زیر به‌روزرسانی می‌شود:

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$$

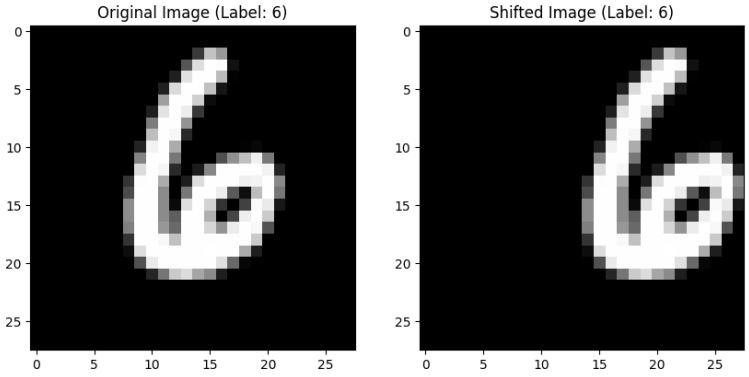
تا زمان همگرایی الگوریتم این روند تکرار خواهد شد.

۳.

الف) ویژگی Transitional in-variance در این شبکه‌ها به این معناست که اشیا فارغ از موقعیت و محل قرارگیری قابل تشخیص هستند.

ب) در لایه‌های Convolution فیتر روی تصویر حرکت کرده و میزان شباهت را با پنجره‌ای از تصویر که روی آن قرار گرفته می‌سنجد در نتیجه اگر جسم وسط یا گوشه تصویر قرار گرفته باشد باز هم توسط این شبکه دیده خواهد شد.
در لایه‌های Pooling خروجی فشرده می‌شود. این امر وابستگی جسم به موقعیت تصویر را کاهش می‌دهد.

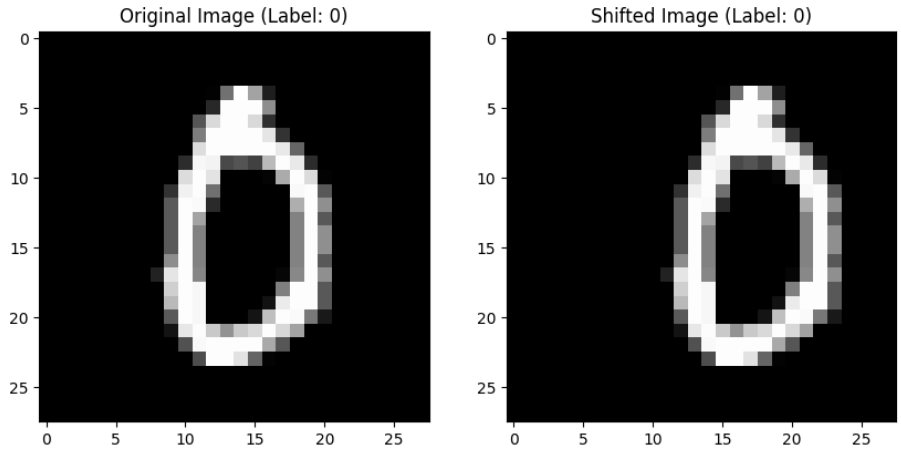
ج) شبکه MLP را در چهار لایه با activation function های ReLU و sigmoid در لایه آخر طراحی می‌کنیم. با train کردن شبکه به دقت 0.9796 می‌رسیم.
با طراحی شبکه LeNet به دقت 0.9842 می‌رسیم. با شیفت دادن عکس‌ها و پیش‌بینی توسط دو مدل می‌بینیم که شبکه LeNet نسبت به تغییر موقعیت عدد در عکس مقاوم است و دچار خطا نمی‌شود.



شکل ۲: تصویر شیفت داده شده در کنار تصویر اصلی

```
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 14ms/step
Predicted label by MLP: 2
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 13ms/step
Predicted label by LeNet: 6
```

شکل ۳: نتایج پیش‌بینی توسط دو مدل



شکل ٤: تصویر شیفت داده شده در کنار تصویر اصلی

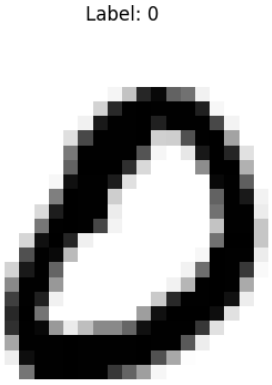
```
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 12ms/step
Predicted label by MLP: 2
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 14ms/step
Predicted label by LeNet: 0
```

شکل ٥: نتایج پیش‌بینی توسط دو مدل

با توجه به نتایج فوق مشاهده می‌کنیم که شبکه MLP تصویر شیفت داده شده را اشتباه پیش‌بینی می‌کند اما شبکه LeNet نسبت به شیفت مقاوم بوده و به درستی نتیجه را پیش‌بینی می‌کند.

.٦

۱- تابع چاپ داده از دیتاست

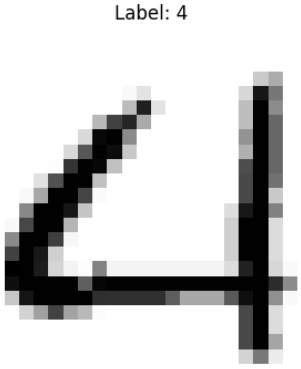


۲- با توجه به نتایج grid search و score بهترین کرنل، کرنل RBF خواهد بود.(نتایج و پارامترهای هر مدل در کد چاپ شده)

۳- با آموزش کرنل RBF روی داده‌های آموزشی خطای آموزشی 0 و خطای تست 0.0869 بدست می‌آید.

٤- خطای ناشی از اعمال Logistic regression روی داده‌های تست 0.1477 بدست می‌آید. این خطا نشان دهنده این است که SVM عملکرد بهتری دارد.

۵-



شکل ٦: داده‌ای که توسط SVM به درستی و توسط Logistic regression به اشتباه طبقه‌بندی شده است