دانشگاه تهران

پردیس دانشکده‌های فنی

دانشکده مهندسی برق و کامپیوتر

تمرین شماره: ۱

یادگیری عمیق با کاربرد در تصویر و صوت

نام و نام‌خانوادگی: مرتضی ملکی‌نژاد شوشتری

شماره دانشجویی: ۸۱۰۱۰۴۲۵۶

نیم‌سال دوم سال تحصیلی ۴۰۵-۴۰۴

فهرست مطالب

۱ محاسبه دستی یک مرحله آموزش شبکه عصبی ۶
۱.۱ محاسبه دستی یک شبکه عصبی ۶
۱.۱.۱ انتشار رو به جلو ۷
۲.۱.۱ انتشار رو به عقب ۸
۲.۱ دلیل استفاده از تابع فعال‌ساز ۱۰
۳.۱ مشکلات سیگموئید ۱۰
۴.۱ نحوه محاسبه گرادیان رلو ۱۰

۲ پیاده‌سازی یک شبکه MLP و مسئله طبقه‌بندی ۱۲
۱.۲ نمونه داده‌های موجود در مجموعه داده ۱۳
۲.۲ توزیع مجموعه داده ۱۳
۳.۲ ابعاد تصاویر ۱۴
۴.۲ تبدیل تصویر به بردار ۱۵
۵.۲ مجموعه‌های آموزش و تست ۱۵
۶.۲ مجموعه اعتبارسنجی ۱۶
۷.۲ مدل پایه ۱۶
۸.۲ نرخ یادگیری ۱۷
۱.۸.۲ جمع بندی ۱۹
۹.۲ استانداردسازی ورودی ۱۹
۱۰.۲ اندازه بسته ۲۱
۱.۱۰.۲ جمع‌بندی ۲۲

۱۱.۲ گشتاور بهینه‌ساز ۲۲
۱.۱۱.۲ جمع‌بندی ۲۴
۱۲.۲ تعداد نورون‌های شبکه ۲۵
۱.۱۲.۲ مقدمه ۲۵
۲.۱۲.۲ نتایج ۲۵
۳.۱۲.۲ جمع‌بندی ۲۶
۱۳.۲ تعداد لایه‌های پنهان ۲۶
۱.۱۳.۲ مقدمه ۲۶
۲.۱۳.۲ نتایج ۲۷
۳.۱۳.۲ جمع‌بندی ۲۸
۱۴.۲ بیش‌برازش ۲۹
۱.۱۴.۲ مقدمه ۲۹
۲.۱۴.۲ نتایج ۳۰
۳.۱۴.۲ جمع‌بندی ۳۱
۱۵.۲ صحت و ماتریس درهم‌ریختگی ۳۲
۱۶.۲ صحت مدل به ازای هر کلاس ۳۳
۱۷.۲ کلاس با بیشترین اشتباه ۳۵
۱۸.۲ کلاس‌هایی که با یکدیگر اشتباه گرفته می‌شوند ۳۵
۱۹.۲ نمونه‌های اشتباه ۳۶

فهرست تصاویر

۱.۱ معماری شبکه ذکر شده در سوال. گره‌های آبی ورودی، گره‌های سبز لایه‌های پنهان و گره قرمز لایه خروجی است. ۷

۱.۲ نمونه عکس‌های موجود در مجموعه داده. از هر کلاس پنج نمونه آورده شده است. ۱۳

۲.۲ توزیع تعداد نمونه‌های هر کلاس در هر بخش که با تعداد کل بخش نرمال شده است. . . ۱۴

۳.۲ در دو نمودار بالا نمودار میزان خطا و دقت مدل روی مجموعه‌های آموزش و اعتبارسنجی در هر دوره رسم شده و در پایین ماتریس درهم‌ریختگی روی هر مجموعه رسم شده‌است. . ۱۷

۴.۲ نمودارهای خطا و دقت و ماتریس درهم‌ریختگی برای ضریب یادگیری $lr = 1$ ۱۸

۵.۲ نمودارهای خطا و دقت و ماتریس درهم‌ریختگی برای ضریب یادگیری $lr = 0.01$ . . ۱۸

۶.۲ نمودارهای خطا و دقت و ماتریس درهم‌ریختگی برای ضریب یادگیری $lr = 1e - 5$. ۱۹

۷.۲ نمودارهای خطا و دقت و ماتریس درهم‌ریختگی در حالتی که ورودی‌ها استاندارد شده باشند. ۲۰

۸.۲ نمودارهای خطا و دقت و ماتریس درهم‌ریختگی در حالتی که اندازه بسته برابر ۸ باشد. . ۲۱

۹.۲ نمودارهای خطا و دقت و ماتریس درهم‌ریختگی در حالتی که اندازه بسته برابر ۱۲۸ باشد. ۲۲

۱۰.۲ نمودارهای خطا و دقت و ماتریس درهم‌ریختگی در حالتی که ضریب گشتاور برابر 0.9 باشد. ۲۳

۱۱.۲ نمودارهای خطا و دقت و ماتریس درهم‌ریختگی در حالتی که ضریب گشتاور برابر 0.7 باشد. ۲۴

۱۲.۲ نمودارهای خطا و دقت و ماتریس درهم‌ریختگی در حالتی که تعداد نورون لایه پنهان برابر ۸ باشد ۲۵

۱۳.۲ نمودارهای خطا و دقت و ماتریس درهم‌ریختگی در حالتی که تعداد نورون لایه پنهان برابر ۶۴ باشد. ۲۶

۱۴.۲ نمودارهای خطا و دقت و ماتریس درهم‌ریختگی در حالتی که مدل دو لایه پنهان با سایزهای ۱۶ و ۳۲ داشته باشد. ۲۸

۱۵.۲ نمودار خطای آموزش و آزمون در حالات مختلف. ۳۱
۱۶.۲ ماتریس درهم‌ریختگی مدل نهایی برای مجموعه داده آزمون. ۳۳
۱۷.۲ نمونه تصاویر رقم ۹ که به اشتباه دسته‌بندی شده‌اند. ۳۵
۱۸.۲ نمونه تصاویری که اشتباه دسته‌بندی شده‌اند. ۳۶

فهرست جداول

۱.۲ انحراف معیار نسبت تعداد نمونه کلاس‌ها برای هر بخش ۱۴
۲.۲ تاثیر استانداردسازی روی مجموعه داده اعتبارسنجی ۲۱
۳.۲ تاثیر اندازه بسته بر دقت مدل در دسته‌بندی مجموعه داده اعتبارسنجی ۲۲
۴.۲ تاثیر گشتاور بر دقت مدل در دسته‌بندی مجموعه داده اعتبارسنجی ۲۴
۵.۲ دقت مدل‌ها با تعداد نورون‌های مختلف در لایه پنهان ۲۶
۶.۲ دقت مدل با دولایه پنهان نسبت به مدل با یک لایه پنهان ۲۸
۷.۲ مقایسه دقت مدل‌ها با روش‌های مختلف جلوگیری از بیش‌برازش ۳۲
۸.۲ دقت مدل نهایی روی مجموعه‌های آموزش، اعتبارسنجی و آزمون ۳۲
۹.۲ دقت، یادآوری و امتیاز F_1 برای هر کلاس ۳۴
۱۰.۲ رتبه‌بندی کلاس‌ها بر اساس دقت کلاسی و یادآوری (صعودی) ۳۴

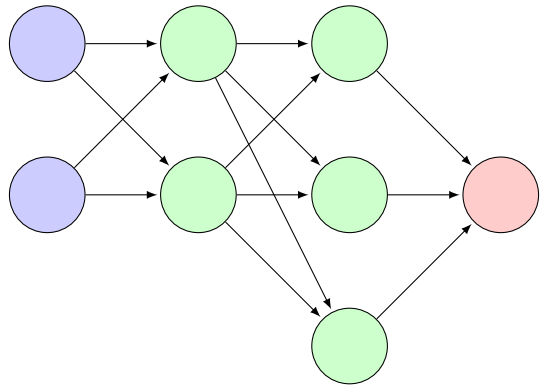
سوال ۱

محاسبه دستی یک مرحله آموزش شبکه عصبی

این سوال به محاسبات مورد نیاز در یک شبکه عصبی و همچنین تاثیر بخش‌های مختلف یک شبکه بر خروجی می‌پردازد.

۱.۱ محاسبه دستی یک شبکه عصبی

در این بخش باید قسمت رو به جلو و رو به عقب یک شبکه MLP حساب شود. معماری شبکه در شکل ۱.۱ رسم شده است.



شکل ۱.۱: معماری شبکه ذکر شده در سوال. گره‌های آبی ورودی، گره‌های سبز لایه‌های پنهان و گره قرمز لایه خروجی است.

اطلاعات شبکه نیز بصورت زیر است (برای مقدار b_2 چون فقط یک عدد صفر در سوال آمده است، فرض شده است آن مقدار برای همه نورون ها است):

$$x = \begin{bmatrix} 1 \\ 0.5 \end{bmatrix} w_1 = \begin{bmatrix} 1 & 0 \\ -1 & 1 \end{bmatrix} b_1 = \begin{bmatrix} 0 \\ 0.5 \end{bmatrix} w_2 = \begin{bmatrix} 0 & -2 & -1 \\ 1 & 1 & 1 \end{bmatrix} b_2 = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} w_3 = \begin{bmatrix} 1 \\ -1 \\ 1 \end{bmatrix} b_3 = \begin{bmatrix} 2 \end{bmatrix} learning_rate = 0.5$$

۱.۱.۱ انتشار رو به جلو

داریم :

$$z_1 = w_1^T x + b_1 = \begin{bmatrix} 1 & -1 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 0.5 \end{bmatrix} + \begin{bmatrix} 0 \\ 0.5 \end{bmatrix} = \begin{bmatrix} 0.5 \\ 1 \end{bmatrix} \quad a_1 = \sigma(z_1) \approx \begin{bmatrix} 0.62 \\ 0.73 \end{bmatrix}$$

$$z_2 = w_2^T a_1 + b_2 = \begin{bmatrix} 0 & 1 \\ -2 & 1 \\ -1 & 1 \end{bmatrix} \begin{bmatrix} 0.62 \\ 0.73 \end{bmatrix} = \begin{bmatrix} 0.73 \\ -0.51 \\ 0.11 \end{bmatrix} \quad a_2 = \sigma(z_2) = \begin{bmatrix} 0.67 \\ 0.38 \\ 0.52 \end{bmatrix}$$

$$z_3 = w_3^T a_2 + b_3 = \begin{bmatrix} 1 & -1 & 1 \end{bmatrix} \begin{bmatrix} 0.67 \\ 0.38 \\ 0.52 \end{bmatrix} + \begin{bmatrix} 2 \end{bmatrix} = \begin{bmatrix} 2.81 \end{bmatrix} a_3 = z_3 = 2.81$$

$$\Rightarrow \hat{y} = 2.81$$

۲.۱.۱ انتشار رو به عقب

اول باید مقدار خطا را محاسبه کرد و سپس گرادیان‌ها را حساب کرد. چون از SGD استفاده می‌شود، فقط باید همین نمونه را در نظر گرفت. داریم:

$$E = error_{MSE} = (y - \hat{y})^2 = (3.97 - 2.81)^2 = (1.16)^2 \approx 1.35$$

$$\frac{\partial E}{\partial \hat{y}} = 2(\hat{y} - y) = -2.32$$

$$\frac{\partial \hat{y}}{\partial a_3} = \frac{\partial a_3}{\partial z_3} = 1$$

$$\delta_i = \frac{\partial E}{\partial z_i} = \frac{\partial E}{\partial z_{i+1}} * \frac{\partial z_{i+1}}{\partial a_i} * \frac{\partial a_i}{\partial z_i} = W_{i+1}\delta_{i+1} \odot \sigma^{'}(z_i)$$

$$\frac{\partial E}{\partial b_i} = \delta_i \quad \frac{\partial E}{\partial W_i} = (a_{i-1}\delta_i^T) \text{ or } (x\delta_i^T)$$

پس داریم:

$$\delta_3 = \frac{\partial E}{\partial z_3} = \begin{bmatrix} -2.32 \end{bmatrix} \qquad \frac{\partial E}{\partial b_3} = -2.32 \qquad \frac{\partial E}{\partial W_3} \approx \begin{bmatrix} -1.55 \\ -0.88 \\ -1.21 \end{bmatrix}$$

$$\delta_2 = W_3\delta_3 \odot \sigma^{'}(z_2) = \begin{bmatrix} 1 \\ -1 \\ 1 \end{bmatrix} \begin{bmatrix} -2.32 \end{bmatrix} \odot \sigma^{'}(\begin{bmatrix} 0.73 \\ -0.51 \\ 0.11 \end{bmatrix}) = \begin{bmatrix} -2.32 \\ 2.32 \\ -2.32 \end{bmatrix} \odot \begin{bmatrix} 0.22 \\ 0.23 \\ 0.25 \end{bmatrix} =$$

$$\begin{bmatrix} -0.51 \\ 0.53 \\ -0.58 \end{bmatrix}$$

$$\frac{\partial E}{\partial b_2} = \begin{bmatrix} -0.51 \\ 0.53 \\ -0.58 \end{bmatrix} \qquad \frac{\partial E}{\partial W_2} = \begin{bmatrix} 0.62 \\ 0.73 \end{bmatrix} \begin{bmatrix} -0.51 & 0.53 & -0.58 \end{bmatrix} = \begin{bmatrix} -0.32 & 0.33 & -0.36 \\ -0.37 & 0.39 & -0.42 \end{bmatrix}$$

$$\delta_1 = W_2 \delta_2 \odot \sigma^{'}(z_1) = \begin{bmatrix} 0 & -2 & -1 \\ 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} -0.51 \\ 0.53 \\ -0.58 \end{bmatrix} \odot \begin{bmatrix} 0.24 \\ 0.2 \end{bmatrix} = \begin{bmatrix} -0.48 \\ -0.56 \end{bmatrix} \odot \begin{bmatrix} 0.24 \\ 0.2 \end{bmatrix} =$$

$$\begin{bmatrix} -0.12 \\ -0.11 \end{bmatrix}$$

$$\frac{\partial E}{\partial b_1} = \begin{bmatrix} -0.12 \\ -0.11 \end{bmatrix} \qquad \frac{\partial E}{\partial w_1} = \begin{bmatrix} 1 \\ 0.5 \end{bmatrix} \begin{bmatrix} -0.12 & -0.11 \end{bmatrix} = \begin{bmatrix} -0.12 & -0.11 \\ -0.06 & -0.06 \end{bmatrix}$$

حال که مشتق خطا نسبت به هرکدام از پارامترها را داریم، باید مقادیر گرادیان‌ها را در ضریب یادگیری ضرب کرد و با مقدار پارامترها جمع کرد. پس داریم:

$$b_{3_{new}} = \begin{bmatrix} 2 \end{bmatrix} - \begin{bmatrix} -1.16 \end{bmatrix} = \begin{bmatrix} 3.16 \end{bmatrix}$$

$$w_{3_{new}} = \begin{bmatrix} 1 \\ -1 \\ 1 \end{bmatrix} - \begin{bmatrix} -0.78 \\ -0.44 \\ -0.6 \end{bmatrix} = \begin{bmatrix} 1.78 \\ -0.56 \\ 1.6 \end{bmatrix}$$

$$b_{2_{new}} = b_2 - \begin{bmatrix} -0.26 \\ 0.27 \\ -0.29 \end{bmatrix} = \begin{bmatrix} 0.26 \\ -0.27 \\ 0.29 \end{bmatrix}$$

$$w_{2_{new}} = \begin{bmatrix} 0 & -2 & -1 \\ 1 & 1 & 1 \end{bmatrix} - \begin{bmatrix} -0.16 & 0.17 & -0.18 \\ -0.19 & 0.2 & -0.21 \end{bmatrix} = \begin{bmatrix} 0.16 & -2.17 & -0.82 \\ 1.19 & 0.8 & 0.79 \end{bmatrix}$$

$$b_{1_{new}} = \begin{bmatrix} 0 \\ 0.5 \end{bmatrix} - \begin{bmatrix} -0.06 \\ -0.06 \end{bmatrix} = \begin{bmatrix} 0.06 \\ 0.56 \end{bmatrix}$$

$$w_{1_{new}} = \begin{bmatrix} 1 & 0 \\ -1 & 1 \end{bmatrix} - \begin{bmatrix} -0.06 & -0.06 \\ -0.03 & -0.03 \end{bmatrix} = \begin{bmatrix} 1.06 & 0.06 \\ -0.97 & 1.03 \end{bmatrix}$$

۲.۱ دلیل استفاده از تابع فعال‌ساز

اگر از تابع فعال‌ساز استفاده نشود و یا تابع فعال‌ساز یک تابع خطی باشد، عملا استفاده از لایه‌های بیشتر و شبکه عمیق‌تر فرقی با یک شبکه تک لایه نخواهد داشت.

$$z_2 = W_2 a_1 + b_2 = W_2(W_1 x + b_1) + b_2 = W_2 W_1 x + W_2 b_1 + b_2 \tag{۱.۱}$$

در فرمول (۱.۱) می‌توان دید اگر تابع فعال‌ساز نباشد، این شبکه دولایه عملا معادل با یک شبکه تک لایه است که در آن w_1 برابر با $w_2 w_1$ در شبکه اصلی است و b_1 برابر با $w_2 b_1 + b_2$ در شبکه اصلی است.

۳.۱ مشکلات سیگموئید

۱. چون مشتق سیگوئید [1] در نواحی دور از صفر به صفر میل می‌کند و مقدار آن بسیار کم است، عملا به روزرسانی وزن‌ها با سرعت کمی انجام می‌شود و زمان آموزش مدل زیاد می‌شود.

۲. چون خروجی بین صفر و یک است و مشتق نیز همواره مثبت است، در به روزرسانی وزن‌ها، گرادیان‌ها هم‌علامت می‌شوند. این موضوع باعث می‌شود که دقت مدل به شدت به وزن‌دهی اولیه وابسته شود و اگر وزن‌دهی اولیه خوب نباشد چون گرادیان‌ها نیز هم‌علامت هستند، به روزرسانی باعث دور شدن از جواب شود (حداقل در آن لایه)

۴.۱ نحوه محاسبه گرادیان رلو

تابع رلو [2] یک تابع دوضابطه‌ای است. پس برای مشتق گرفتن از آن باید مشتق هر ضابطه را گرفت و مشتق تابع رلو در بازه ضابطه برابر با مشتق آن ضابطه خواهد بود.

$$\sigma(x) = \begin{cases} 0 & (x \leq 0) \\ x & (x > 0) \end{cases} \tag{۲.۱}$$

[1] Sigmoid
[2] ReLU

تعریف تابع رلو در (۲.۱) آمده است. با مشتق گرفتن از هر ضابطه و درنظر گرفتن شرط آن می‌توان گفت:

$$\sigma^{'}(x) = \begin{cases} 0 & (x \leq 0) \\ 1 & (x > 0) \end{cases} = U(x) \tag{۳.۱}$$

که این همان تعریف تابع پله است.

سوال ۲

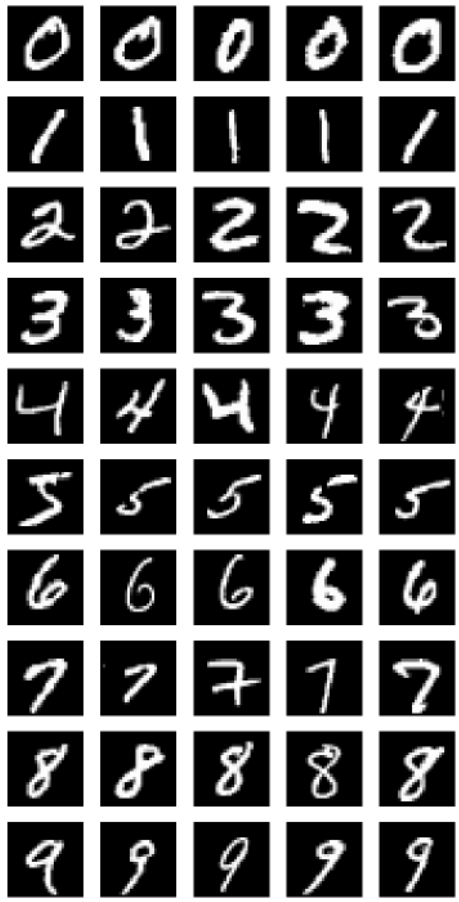
پیاده‌سازی یک شبکه MLP و مسئله طبقه‌بندی

در این سوال باید مجموعه داده MNIST بررسی شود و سپس برای طبقه‌بندی آن یک شبکه عصبی از ابتدا و فقط با استفاده از numpy برای محاسبات، پیاده‌سازی شود و سپس تاثیر بخش‌های مختلف شبکه روی خروجی بررسی شود.

آماده‌سازی داده

(سیر اجرای همه بخش‌های این قسمت در نوتبوک `exploration` آمده است.)

۱.۲ نمونه داده‌های موجود در مجموعه داده

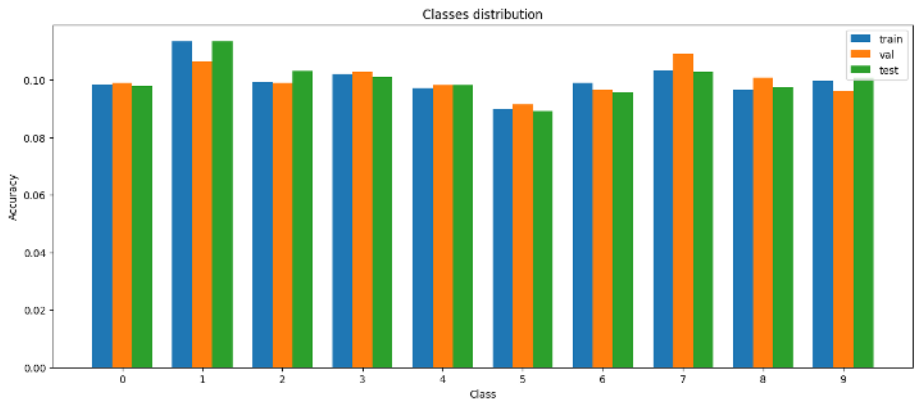


شکل ۱.۲: نمونه عکس‌های موجود در مجموعه داده. از هر کلاس پنج نمونه آورده شده است.

۲.۲ توزیع مجموعه داده

مجموعه داده به سه بخش $50,000$ ، $10,000$ و $10,000$ تقسیم شده است که به ترتیب برای آموزش ، اعتبارسنجی و آزمون در نظر گرفته شدند. در شکل ۲.۲ توزیع کلاس‌های هر بخش آمده است. در سوال هیستوگرام خواسته شده که با توجه به تعداد کم کلاس‌ها، نیازی به درنظرگرفتن دسته [1] نیست و می‌توان با نمودار میله‌ای اطلاعات کامل‌تری به‌دست آورد.

[1] bin



شکل ۲.۲: توزیع تعداد نمونه‌های هر کلاس در هر بخش که با تعداد کل بخش نرمال شده است.

همان‌طور که در شکل ۲.۲ مشاهده شد، تعداد نمونه‌های هر کلاس تقریبا ده درصد کل داده‌های آن بخش را تشکیل می‌دهد و به نظر ناترازی خاصی وجود ندارد. برای بررسی بیشتر انحراف معیار تعداد دسته‌ها حساب شد که در جدول ۱.۲ نشان داده شده است. با توجه به اینکه مقادیر انحراف معیار بسیار کوچک و نزدیک به صفر است، پس دوباره به این می‌توان رسید که ناترازی خاصی وجود ندارد.

جدول ۱.۲: انحراف معیار نسبت تعداد نمونه کلاس‌ها برای هر بخش

بخش	انحراف معیار
train	0.005648
validation	0.004846
test	0.00592

۳.۲ ابعاد تصاویر

در این بخش خواسته شده تا ابعاد تصاویر بررسی شود تا همه ابعاد 28×28 را داشته باشند. همانطور که در تکه کد ۱ و خروجی آن آمده است، می‌توان دید که تصاویر به‌صورت یک بردار با سایز ۷۸۴ نشان داده شده‌اند. و تصویری وجود ندارد که بعد متفاوتی داشته باشد و نیاز به تبدیل آن باشد.

```
from data.data_loader import load_data
data = load_data()
for key, val in data.__dict__.items():
    print(key, val.shape)
```

```
X_train (50000, 784)
y_train (50000, 10)
X_val (10000, 784)
y_val (10000, 10)
X_test (10000, 784)
y_test (10000, 10)
```

نمونه کد ۱: ابعاد آرایه‌های مختلف مجموعه داده

برای تبدیل آن‌ها به یک عکس 28×28 می‌توان مشابه کد ۲ عمل کرد. البته این تبدیل فقط برای نمایش عکس بکار می‌رود و برای شبکه عصبی همان بردار بکار می‌رود.

```
IMSIZE = (28, 28)
img = data.X_train[0]
img = img.reshape(IMSIZE)
```

نمونه کد ۲: ابعاد آرایه‌های مختلف مجموعه داده

۴.۲ تبدیل تصویر به بردار

خود تصاویر در ابتدا به سایز بردارهایی با سایز ۷۸۴ هستند. از آنجایی که تصاویر به‌صورت بردار ذخیره شده‌اند، نیازی به تغییر ورودی نیست و همان‌طور که در سوال نیز ذکر شده ورودی شبکه برداری با ۷۸۴ مقدار است.

۵.۲ مجموعه‌های آموزش و تست

اصل هدف در طبقه‌بندی توانایی طبقه‌بندی داده‌های جدید است که مدل تا به حال آن‌ها را ندیده است. پس نیاز است تا مقداری از داده‌ها در فرآیند آموزش استفاده نشوند و برای بررسی توانایی تعمیم‌پذیری [۲] مدل بکار روند تا تخمینی از میزان دقت مدل روی داده‌های واقعی که قرار است با آن‌ها سر و کار داشته باشد داشته باشیم.

همچنین چیزی که مهم است این است که از این مجموعه آزمون در هیچ جای فرآیند آموزش، حتی مثلا در محاسبه پارامترهای مربوط به استانداردسازی داده استفاده نشود. اگر این اتفاق صورت نگیرد به اصطلاح

[2] Generalization

نشتی [3] رخ داده است و عملا صحت نتایج مدل زیر سوال می‌رود. پس بهتر است این جداسازی در همان مراحل اولیه و پس از یک تصویرسازی [4] ساده از توزیع داده‌ها و زمانی که هیچ تلاشی برای آموزش مدل صورت نگرفته انجام شود.

۶.۲ مجموعه اعتبارسنجی

در همین مجموعه داده از قبل ۱۰۰۰۰ عضو از ۶۰۰۰۰ داده آموزش جدا شده اند و نیازی به جداسازی مجدد نیست. مجموعه داده اعتبارسنجی وقتی مورد نیاز است که برای یک مسئله چندین مدل آموزش داده شده‌اند و نیاز است تا بهترین مدل انتخاب شود.

قاعدتا دقت مدل روی مجموعه داده آموزش معیار مناسبی برای انتخاب نیست زیرا از مهم ترین مسائل دنیای یادگیری ماشین همین تعمیم پذیری است. از طرفی اگر از مجموعه داده آزمون استفاده شود، به نوعی نشتی رخ داده. زیرا مجموعه داده آزمون که باید هیچ نقشی در آموزش نداشته باشد، در انتخاب مدل دخیل شده است. به همین منظور یک مجموعه داده مجزا برای مقایسه مدل‌ها قبل از انتخاب مدل نهایی استفاده می‌شود و از مجموعه آزمون فقط برای تخمین دقت روی داده واقعی استفاده می‌شود.

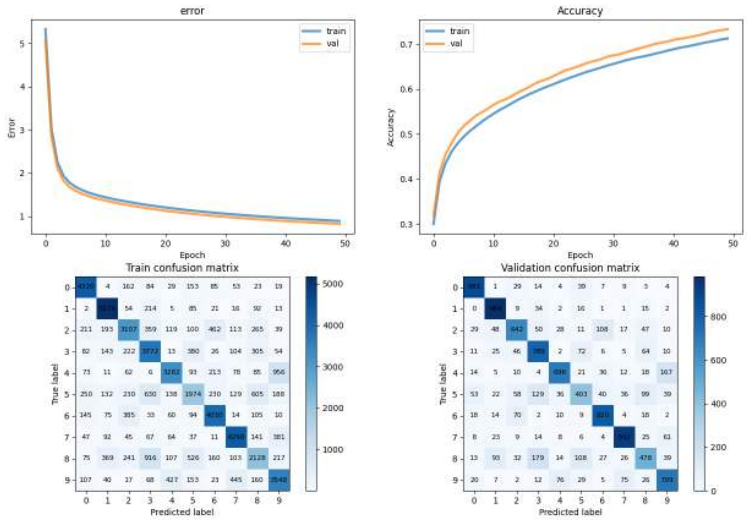
آموزش و ارزیابی مدل

(سیر اجرای همه بخش‌های این قسمت در نوتبوک `run_on_mnist` آمده است.)

۷.۲ مدل پایه

همان‌طور که در شکل ۳.۲ مشاهده می‌شود، هنوز مدل جای بهبود دارد و بهتر است تعداد دوره‌ها زیاد شود زیرا همچنان خطا کاهشی است و بین خطای آموزش و اعتبارسنجی تفاوت زیادی نیست (در حقیقت خطای مجموعه اعتبارسنجی حتی کمی کمتر است!)

[3] Leakage
[4] Visualization



شکل ۳.۲: در دو نمودار بالا نمودار میزان خطا و دقت مدل روی مجموعه‌های آموزش و اعتبارسنجی در هر دوره رسم شده و در پایین ماتریس درهم‌ریختگی روی هر مجموعه رسم شده‌است.

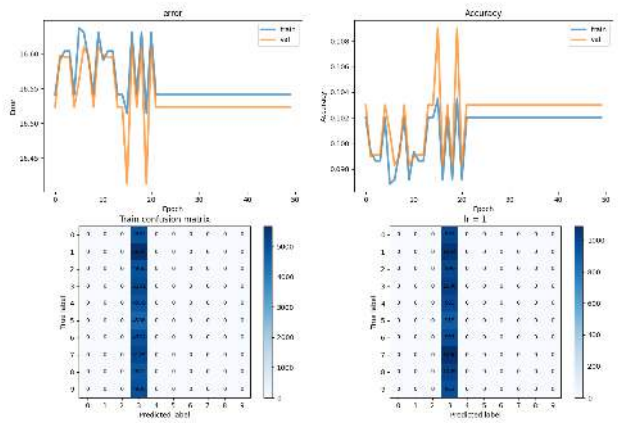
همچنین با مشاهده ماتریس درهم ریختگی می‌توان دید که برای مثال در تشخیص رقم ۸، تعداد زیادی از عکس‌ها به اشتباه به ۳ طبقه بندی شده‌اند و پس از تنظیمات کلی و عمومی مثل ضریب یادگیری می‌توان روی این مورد تمرکز کرد و برای مثال یا داده بیشتری تهیه کرد یا از افزایش مصنوعی داده‌ها [۵] برای تولید نمونه‌های بیشتری از رقم ۸ بهره برد.

۸.۲ نرخ یادگیری

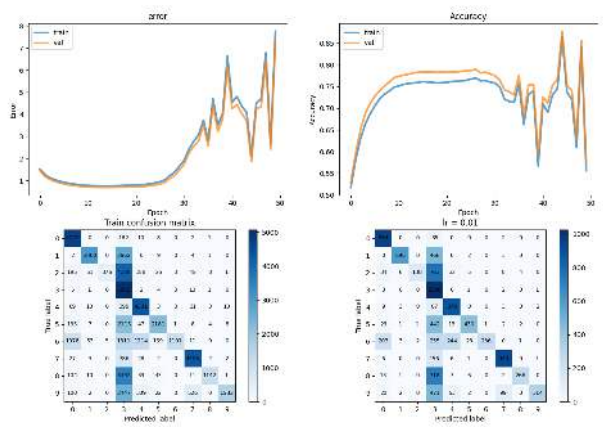
- **نرخ یادگیری کم:** با نرخ یادگیری بسیار کم، از آنجایی که در هر مرحله به روز رسانی، وزن‌ها تغییر کمی می‌کنند؛ مدل خیلی کند به سمت نقطه بهینه حرکت می‌کند و فرآیند آموزش طولانی می‌شود. اگر خطای مدل همواره کاهشی است ولی سرعت آن کم است این احتمال وجود دارد که نرخ یادگیری را باید زیاد کرد.
- **نرخ یادگیری زیاد:** با نرخ یادگیری زیاد، این‌که در مراحل اولیه ممکن و محتمل است که خطا با سرعت خوبی کاهش یابد ولی چون وزن‌ها تغییر زیادی می‌کنند و گام بزرگ است می‌تواند باعث شود

[5] Data Augmentation

مدل هیچوقت نتواند به نقطه بهینه برسد و صرفا دور و بر آن نوسان کند. اگر در هنگام آموزش خطا مدام کم و زیاد شود، ممکن است به این معنی باشد که باید نرخ یادگیری کمتر شود. البته باید این موضوع روی خطای کل مجموعه بررسی شود نه خطای یک دسته [۶]

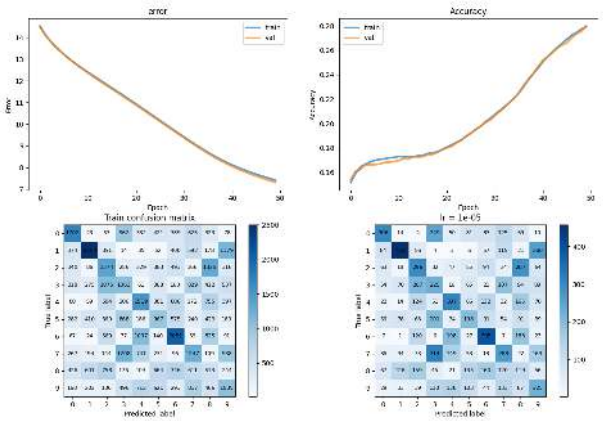


شکل ۴.۲: نمودارهای خطا و دقت و ماتریس درهم‌ریختگی برای ضریب یادگیری $lr = 1$



شکل ۵.۲: نمودارهای خطا و دقت و ماتریس درهم‌ریختگی برای ضریب یادگیری $lr = 0.01$

[6] Batch



شکل ۶.۲: نمودارهای خطا و دقت و ماتریس درهم‌ریختگی برای ضریب یادگیری $lr = 1e-5$

با توجه به نمودارهای اشکال ۴.۲ ، ۵.۲ و ۶.۲ می‌توان دید وقتی ضریب یادگیری برابر یک باشد، مدل همگرا نمی‌شود و عملا به سمت نقطه خاصی نمی‌رود. وقتی ضریب برابر 0.01 باشد نیز با اینکه در ابتدا کمی خطا کم می‌شود ولی مدل به دور نقطه بهینه نوسان می‌کند و مقدار خطا افزایش می‌یابد. به دلیل گام‌های بزرگ، فرآیند آموزش ناپایدار شده و مدل قادر به همگرایی پایدار نیست. وقتی ضریب یادگیری بسیار کم باشد می‌توان دید که مدل در حال بهبود است ولی سرعت این بهبود بسیار کم است به طوری که در انتهای دور پنجاهم همچنان مدل به دقت مدل پایه (با ضریب یادگیری 0.001) بعد از یک دور نرسیده است.

۱.۸.۲ جمع بندی

با توجه به نتایج بدست آمده، همان مقدار $1e-3$ در مدل پایه از باقی مقادیر مناسب‌تر است.

۹.۲ استانداردسازی ورودی

نرمال‌سازی یعنی داده‌ها به یک مقیاس مشخص (معمولا صفر تا یک) برده شوند. از فرمول (۱.۲) برای آن استفاده می‌شود.

$$x^{'} = \frac{x - x_{min}}{x_{max} - x_{min}} \tag{۱.۲}$$

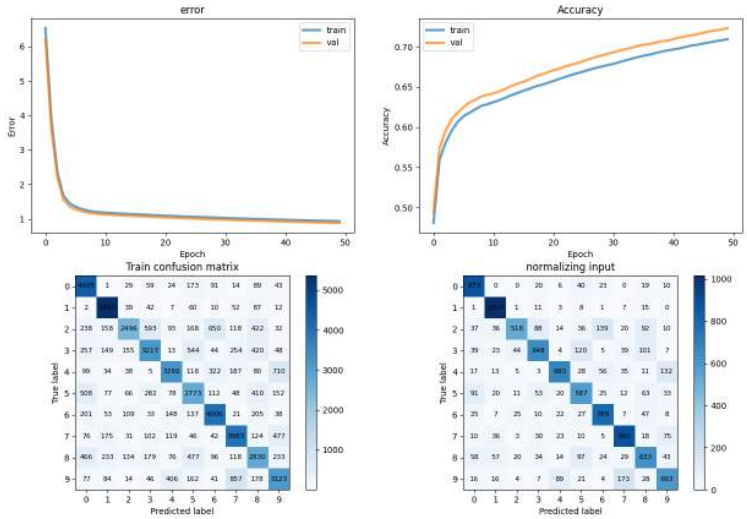
با این‌کار امید می‌رود تا توزیع ویژگی‌های مختلف به هم نزدیک شود که در شبکه عصبی مهم است زیرا اگر توزیع‌ها خیلی باهم متفاوت باشند، شبکه باید با دادن وزن‌های بسیار بالا یا بسیار کم این مسئله را جبران کند که باعث کندی آموزش و یا سرریز [7] می‌شود.

از طرف دیگر در استانداردسازی عملا فرض می‌شود که توزیع‌ها نرمال هستند و فقط آن‌ها به نرمال استاندارد تبدیل می‌شوند. فرض نرمال بودن فرض عجیبی نیست زیرا از یک طرف طبق قضیه حد مرکزی و این موضوع که این ویژگی خود حاصل تاثیر چندین مورد دیگر است احتمال نرمال بودن توزیع بالاست و حتی اگر نرمال هم نباشد، خنثی کردن اثر میانگین (bias ذاتی ویژگی) و پراکندگی توزیع‌ها را فارغ از نرمال بودن به هم نزدیک می‌کند.

در استاندارد سازی از فرمول (۲.۲) استفاده می‌شود.

$$x^{'} = \frac{x - \bar{x}}{\sigma_x} \tag{۲.۲}$$

در ادامه این سوال از استانداردسازی استفاده می‌شود و نرمال‌سازی استفاده نمی‌شود.



شکل ۷.۲: نمودارهای خطا و دقت و ماتریس درهم‌ریختگی در حالتی که ورودی‌ها استاندارد شده باشند.

[7] overflow

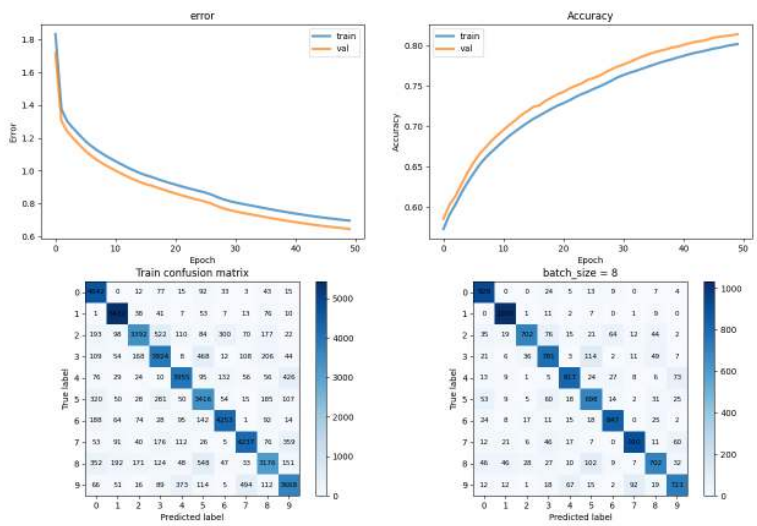
همان‌طور که در شکل ۷.۲ مشاهده می‌شود، عملکرد مدل اندکی بهبود داشته و از این قسمت به بعد از استانداردسازی استفاده می‌شود. نتایج نهایی مدل‌ها در جدول ۲.۲ آمده است.

جدول ۲.۲: تاثیر استانداردسازی روی مجموعه داده اعتبارسنجی

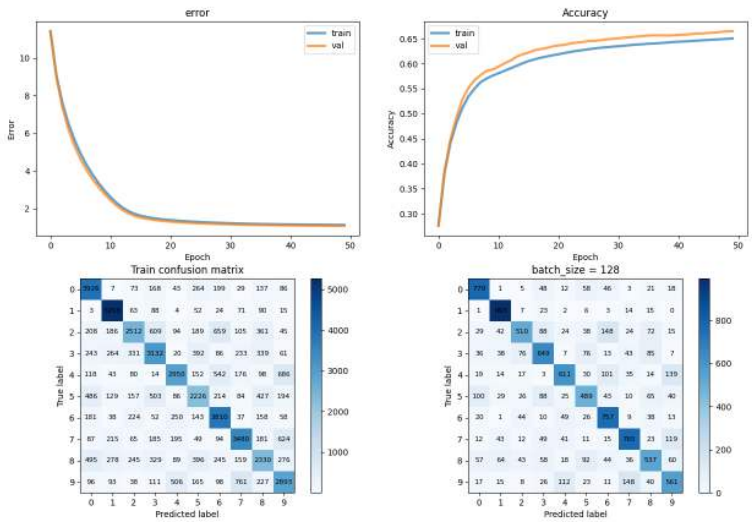
بدون استانداردسازی	0.66616
با استانداردسازی	0.7231

۱۰.۲ اندازه بسته

به طور کلی سایز بسته کوچک‌تر باعث می‌شود همگرایی سریع‌تر ولی پر نویزتر باشد و خطای مدل نوسان بیشتری خواهد داشت. مطابق نمودارهای ۸.۲ و ۹.۲ می‌توان دید که نکته مربوط به بهبود رخ داده ولی ناپایداری‌ای مشاهده نمی‌شود. این به این علت است که مقدار خطا هنگام گزارش دادن روی یک دور کامل از داده حساب می‌شود و نه روی یک دسته.



شکل ۸.۲: نمودارهای خطا و دقت و ماتریس درهم‌ریختگی در حالتی که اندازه بسته برابر ۸ باشد.



شکل ۹.۲: نمودارهای خطا و دقت و ماتریس درهم‌ریختگی در حالتی که اندازه بسته برابر ۱۲۸ باشد.

۱.۱۰.۲ جمع‌بندی

با توجه به جدول ۳.۲ اندازه بسته ۸ بهترین نتیجه را می‌دهد و از این به بعد از این مقدار استفاده می‌شود.

جدول ۳.۲: تاثیر اندازه بسته بر دقت مدل در دسته‌بندی مجموعه داده اعتبارسنجی

اندازه بسته	دقت
۸	0.8142
۳۲	0.7231
۱۲۸	0.6651

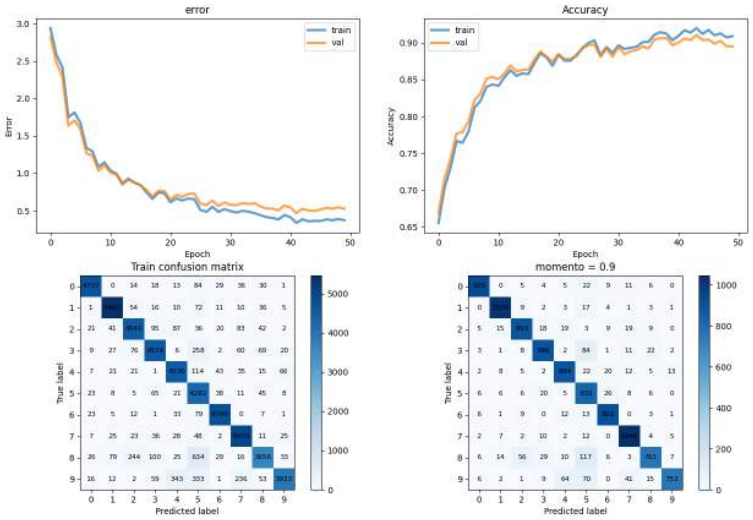
۱۱.۲ گشتاور بهینه‌ساز

اضافه کردن گشتاور باعث می‌شود تا مدل در جاهایی مثل نقاط زینی [۸] یا جایی با مشتق کم [۹] گیر نیفتد و از آن‌ها گذر کند و سریع‌تر همگرا شود.

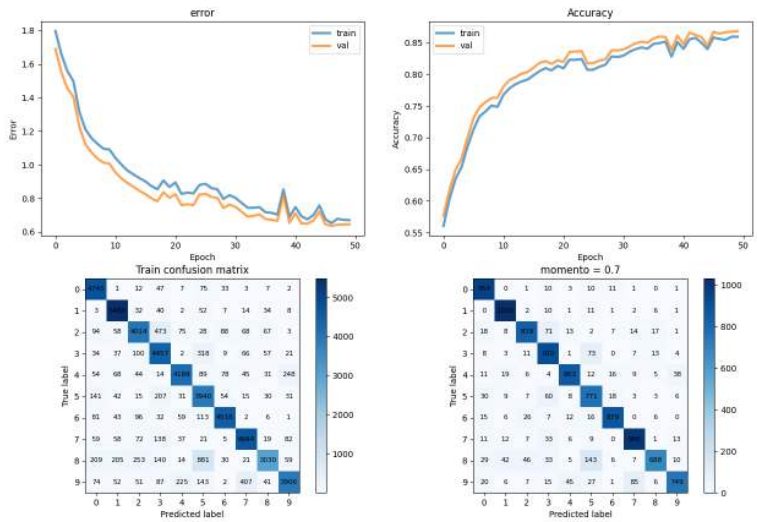
با تنظیم ضریب گشتاور به مقدار 0.9 می‌توان دید که مقدار خطا سریع‌تر کاهش می‌یابد. همچنین می‌توان

[8] Saddle point
[9] Plateau

دید با کاهش خطا از مقدار مشخصی، نمودار خطا و دقت مقداری رفتار نوسانی پیدا می‌کنند که می‌تواند نشانگر این باشد که ضریب یادگیری را باید کاهش داد. ولی چون در ادامه سوال ابرمتغیرهای دیگری نیز تغییر می‌کنند که روی این مسئله اثر می‌گذارند و سوال نخواسته، فعلا از این مورد صرف‌نظر شده است.



شکل ۱۰.۲: نمودارهای خطا و دقت و ماتریس درهم‌ریختگی در حالتی که ضریب گشتاور برابر 0.9 باشد.



شکل ۱۱.۲: نمودارهای خطا و دقت و ماتریس درهم‌ریختگی در حالتی که ضریب گشتاور برابر 0.7 باشد.

۱.۱۱.۲ جمع‌بندی

با توجه به جدول ۴.۲ می‌توان دید که گشتاور تاثیر مثبتی دارد و از این قسمت به بعد از آن استفاده می‌شود. همچنین دو مقدار مختلف برای ضریب آن تست شد که با توجه به ضریب یادگیری و برای جلوگیری از سرریز ضریب 0.9 برای آن انتخاب شد. [۱۰]

جدول ۴.۲: تاثیر گشتاور بر دقت مدل در دسته‌بندی مجموعه داده اعتبارسنجی

نوع مدل	دقت
بدون گشتاور	0.8142
با گشتاور با ضریب 0.9	0.895
با گشتاور با ضریب 0.7	0.8681

[۱۰] برای گزینه‌های مرسوم ضریب گشتاور از هوش مصنوعی کمک گرفته شد.

۱۲.۲ تعداد نورون‌های شبکه

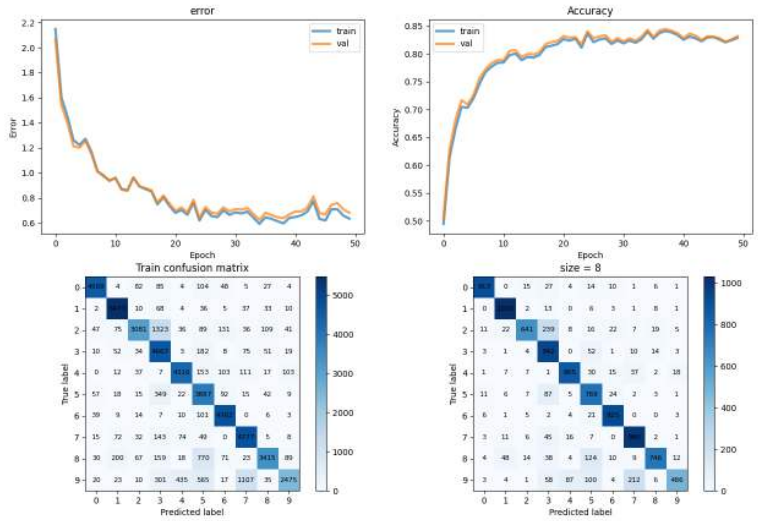
۱.۱۲.۲ مقدمه

از آنجایی که شبکه اولیه بسیار ساده است انتظار می‌رود تا با زیاد کردن تعداد نورون دقت بهبود یابد زیرا در نمودارهای قبلی از آنجایی که اختلاف بین خطای آموزش و اعتبارسنجی زیاد نشده بیش‌برازش نیز رخ نداده.

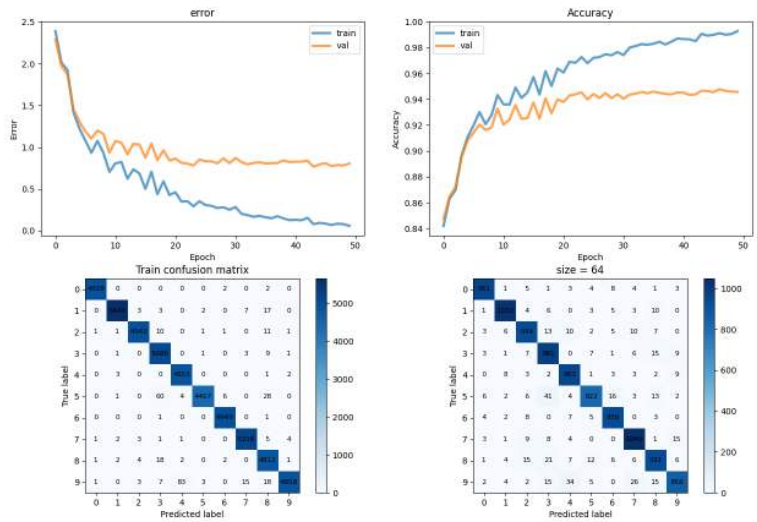
۲.۱۲.۲ نتایج

همان‌طور که انتظار می‌رفت با افزایش نورون‌ها و پیچیده‌تر شدن شبکه، دقت بهبود می‌یابد. ولی مسئله جدیدی که پدید می‌آید مسئله بیش‌برازش است. می‌توان دید هرچقدر شبکه پیچیده‌تر می‌شود اختلاف دقت بین مجموعه آموزش و اعتبارسنجی بیشتر می‌شود.

با مقایسه نمودارهای ۱۲.۲ و ۱۳.۲ می‌توان دید که در مدل بزرگ‌تر این‌که دقت بهبود یافته ولی فاصله بین دو نمودار مربوط به آموزش و اعتبارسنجی بیشتر شده است که حاکی از بیش‌برازش است.



شکل ۱۲.۲: نمودارهای خطا و دقت و ماتریس درهم‌ریختگی در حالتی که تعداد نورون لایه پنهان برابر ۸ باشد



شکل ۱۳.۲: نمودارهای خطا و دقت و ماتریس درهم‌ریختگی در حالتی که تعداد نورون لایه پنهان برابر ۶۴ باشد.

۳.۱۲.۲ جمع‌بندی

طبق جدول ۵.۲ علی‌رغم بیش‌برازش، مدل با ۶۴ نورون روی مجموعه داده اعتبارسنجی نیز عملکرد بهتری دارد و از این قسمت به بعد از آن استفاده می‌شود.

جدول ۵.۲: دقت مدل‌ها با تعداد نورون‌های مختلف در لایه پنهان

تعداد نورون	دقت روی داده آموزش	دقت روی داده اعتبارسنجی
۸	0.82844	0.8316
۱۶	0.90904	0.895
۶۴	0.99278	0.9458

۱۳.۲ تعداد لایه‌های پنهان

۱.۱۳.۲ مقدمه

برای آن‌که حدسی از عملکرد مدل داشته باشیم باید پیچیدگی مدل را بیابیم. در قسمت قبل این‌کار ساده بود زیرا در یک لایه تغییرات انجام می‌شد و عمق مدل ثابت بود و فقط تعداد نورون‌ها تغییر می‌کرد. ولی در این‌جا

باید تعداد پارامتر مدل‌ها را باهم مقایسه کرد.

برای مدل اول (مدل با یک لایه پنهان با ۶۴ نورون) تعداد پارامترها به‌صورت زیر است:

$$parameter\ count = parameter_{hidden} + parameter_{output}$$
$$= (784 * 64 + 64) + (64 * 10 + 10) = 50890$$

برای مدل دوم با دولایه پنهان با سایزهای ۱۶ و ۳۲ داریم:

$$parameter\ count = parameter_{hidden0} + parameter_{hidden1} + parameter_{output}$$
$$= (784 * 16 + 16) + (16 * 32 + 32) + (32 * 10 + 10) = 13434$$

که این تقریبا برابر با تعداد پارامترهای یک مدل با یک لایه پنهان با ۱۶ نورون است.

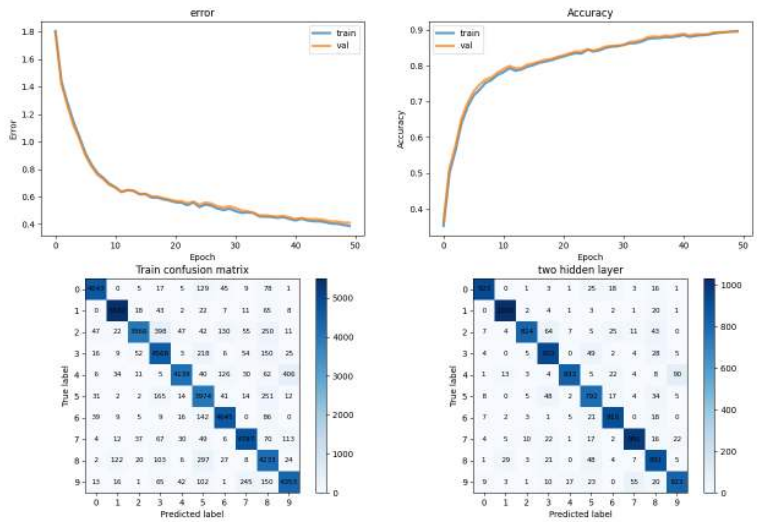
پس یعنی انتظار می‌رود مدل عمیق‌تر از آنجایی که پارامتر کمتری دارد بیش‌برازش نداشته باشد یا دست کم بیش‌برازش کمتر در آن رخ دهد.

از لحاظ دقت مدل، چون پارامترهای مدل تک‌لایه بسیار بیشتر است، انتظار می‌رود تا خطای آموزش کمتری داشته باشد. ولی برای داده اعتبارسنجی نمی‌توان با اطمینان صحبت کرد. زیرا مدل دولایه با آنکه پارامترهای کمتری دارد، ولی چون عمق بیشتری دارد می‌تواند ویژگی‌های پیچیده‌تری را کشف کند و دقت خوبی بدهد. یعنی مدل دولایه حتی با پارامتر کمتر، به دلیل عمق بیشتر می‌تواند قدرت نمایش بالاتری نسبت به یک لایه کم‌نورون داشته باشد.

۲.۱۳.۲ نتایج

همان‌طور که در شکل ۱۴.۲ می‌توان دید، مدل دولایه دچار بیش‌برازش نشده و مقادیر دقت و خطا برای مجموعه آموزش و اعتبارسنجی بسیار نزدیک به‌هم هستند.

از لحاظ دقت، دقت مدل بسیار نزدیک به مدل با ۱۶ نورون در بخش قبل است. البته با توجه به شکل ۱۰.۲ می‌توان دید که در مدل ۱۶ نورون، پس از پنجاه دور آموزش مدل تقریبا اشباع شده و دیگر خطا بهبودی نمی‌یابد. ولی در نمودار مدل دولایه ۱۴.۲ می‌توان دید که خطا همچنان نزولی است و مدل اشباع نشده.



شکل ۱۴.۲: نمودارهای خطا و دقت و ماتریس درهم‌ریختگی در حالتی که مدل دو لایه پنهان با سایزهای ۱۶ و ۳۲ داشته باشد.

۳.۱۳.۲ جمع‌بندی

علی‌رغم این‌که مدل دولایه بیش‌برازش کمتری دارد ولی باز دقت مدل تک‌لایه روی داده اعتبارسنجی از مدل دولایه بهتر بود. البته احتمالا با تنظیم تعداد نورون‌ها و دور آموزش بتوان نتیجه بهتری از مدل‌های عمیق‌تر گرفت که چون خواست سوال نیست و در ادامه سعی می‌شود مسئله بیش‌برازش نیز حل شود، از آن صرف‌نظر می‌شود.

جدول ۶.۲: دقت مدل با دولایه پنهان نسبت به مدل با یک لایه پنهان

مدل	دقت روی داده آموزش	دقت روی داده اعتبارسنجی
تک لایه	0.99278	0.9458
دو لایه	0.895	0.8962

۱۴.۲ بیش‌برازش

۱.۱۴.۲ مقدمه

موقع آموزش داده، ممکن است مدل به جای یادگیری الگوهای تعیین کننده داده، شروع به یادگیری نویز بکند و بتواند دقت خود را روی مجموعه داده آموزش بهبود دهد. ولی از آنجایی که این نویزها تکرار پذیر نیستند (اگر تکرارپذیر بودند، نویز نیستند و یک الگو در تصاویر هستند.)، روی داده‌هایی که مدل تا به حال آن‌ها را ندیده، وجود ندارند. و در نتیجه دقت خوب و خطای پایین مدل روی داده آموزش، روی داده اعتبارسنجی و یا داده آزمون مدل عملکرد خوبی نشان نخواهد داد.

مدل‌هایی که در مراحل قبل تست شدند، عموما مدل‌های خیلی پیچیده‌ای نبودند و برای همین احتمال رخ دادن بیش‌برازش در آن‌ها کم است. تنها مدلی که با این مسئله مواجه شد مدل با ۶۴ نورون بود. همان‌طور که در تصویر ۱۳.۲ می‌توان دید، فاصله بین نمودارهای مربوط به داده آموزش و اعتبارسنجی زیاد است و مدل دچار بیش‌برازش شده است.

راه‌های رفع بیش‌برازش

راه‌های متعددی برای رفع بیش‌برازش وجود دارد. در اینجا دو مورد از آن‌ها که در سوال آمده توضیح داده می‌شوند:

- **منظم‌سازی:** منظم‌سازی [۱۱] به این صورت است که در کنار هدف اصلی مدل که کاهش خطا است، یک هدف دیگر مطرح می‌شود که کم بودن مقدار وزن‌ها است. به عبارت دیگر، مدل بابت مقدار زیاد وزن‌ها جریمه می‌شود. زیرا بیش‌برازش به علت حساسیت زیاد به ورودی است و معمولا به این صورت رخ می‌دهد که مدل از بین همه ویژگی‌های ورودی، تنها به بخش محدودی از آن‌ها توجه کند و به آن‌ها وزن زیادی بدهد. پس اگر جلوی مقادیر بالای وزن‌ها گرفته شود، هر لایه مجبور می‌شود به سایر ویژگی‌ها نیز توجه کند و احتمال رخ دادن بیش‌برازش کمتر می‌شود.

  این کار به این صورت انجام می‌شود که خطای نهایی مدل به‌صورت زیر در نظر گرفته می‌شود (دقت کنید که چون بیش‌برازش از حساسیت بیش از حد به داده به وجود می‌آید و مقدار بایاس ورودی نمی‌گیرد، نیازی نیست در منظم‌سازی لحاظ شود.):

$$L_{total} = L_0 + \lambda \sum_l W_l^2 \tag{۳.۲}$$

[11] Regularization

و موقع انتشار رو به عقب، گرادیان به‌صورت زیر تغییر می‌کند:

$$\nabla W_l = \frac{\partial L_0}{\partial W_l} + 2\lambda W_l \tag{۴.۲}$$

پس یعنی در هر مرحله به روزرسانی از عدد خود وزن نیز مقداری کاسته می‌شود.

البته فرمولی که گفته شده، مربوط به یک منظم‌سازی خاص به اسم منظم‌سازی l2 است. [۱۲] و ممکن است از روابط دیگری استفاده شود.

- **توقف زودهنگام:** در شرایطی ممکن است با آموزش بیشتر مدل نه تنها دقت روی مجموعه اعتبارسنجی بهبود نیابد، بلکه به علت بیش‌برازش دقت کمتر شود. پس می‌توان با تشخیص این مورد، بدون اینکه مدل همه تعداد دوری که مشخص شده آموزش ببیند، آموزش را متوقف کرد و مدلی داشت که تعمیم‌پذیری بهتری دارد.

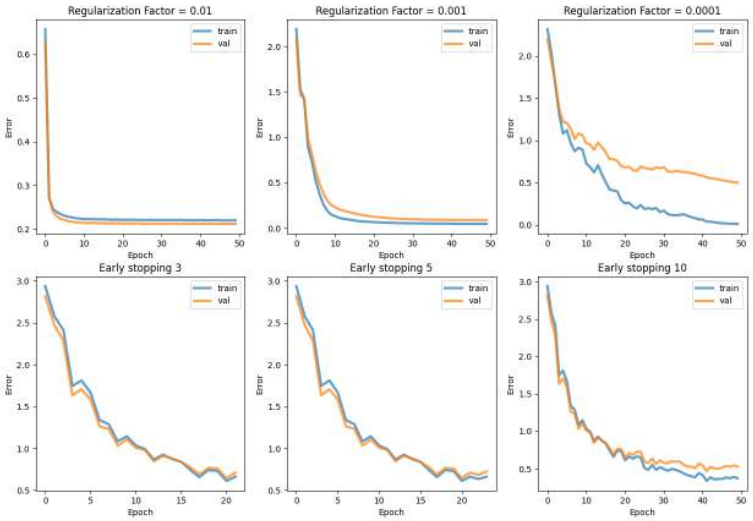
پیاده‌سازی این مورد به این صورت است که اگر مدل برای مثال برای ۱۰ دور، دقتی پایین‌تر روی داده اعتبارسنجی داشته باشد (نسبت به دقت دوره‌های عقب تر) آموزش متوقف می‌گردد.

البته با توجه به نمودار ۱۳.۲ از آنجایی که خطای اعتبارسنجی افزایش نیافته و صرفاً اشباع شده است، توقف زودهنگام در این مسئله احتمالا اثر قابل توجهی نداشته باشد.

۲.۱۴.۲ نتایج

برای توقف زودهنگام حد آستانه‌های سه، پنج و ده امتحان شدند و برای منظم‌سازی برای ضریب منظم‌سازی مقادیر $10^{-3}, 10^{-2}$ و 10^{-4} امتحان شدند.

[12] L2 Regularization



شکل ۱۵.۲: نمودار خطای آموزش و آزمون در حالات مختلف.

همانطور که در شکل ۱۵.۲ می‌توان دید، مطابق انتظار روش توقف زودهنگام خیلی کمکی نکرده و با این‌که تفاوت خطا بین داده آموزش و اعتبارسنجی کاهش یافته ولی این به علت عدم کاهش خطا روی داده آموزش است و خطای داده اعتبارسنجی بهبودی نداشته. حتی در حد آستانه‌های پایین این مورد باعث شده تا دقت مدل روی داده اعتبارسنجی نیز کم شود.

ولی می‌توان دید که با منظم‌سازی با انتخاب ضریب مناسب، مسئله بیش‌برازش تا حد خوبی در این‌جا حل می‌شود. در منظم‌سازی انتخاب ضریب نقش مهمی دارد. همانطور که در شکل ۱۵.۲ می‌توان دید، با یک ضریب کم عملا منظم‌سازی رخ نمی‌دهد و مدل همچنان دچار بیش‌برازش می‌شود. با یک ضریب بالا نیز، وزن‌های مدل نمی‌توانند به مقدار بهینه برسند و مدل آموزش نمی‌بیند.

۳.۱۴.۲ جمع‌بندی

استفاده از منظم‌سازی با ضریب 10^{-3} بهترین نتیجه را می‌دهد و از این روش استفاده می‌شود.

تحلیل نتایج آزمون

جدول ۷.۲: مقایسه دقت مدل‌ها با روش‌های مختلف جلوگیری از بیش‌برازش

روش	پارامتر روش	دقت روی مجموعه آموزش	دقت روی مجموعه اعتبارسنجی
مدل پایه	–	0.90904	0.8950
منظم‌سازی	0.01	0.94514	0.9497
منظم‌سازی	0.001	0.99124	0.9750
منظم‌سازی	0.0001	0.99616	0.9492
توقف زودهنگام	3	0.87574	0.8782
توقف زودهنگام	5	0.88362	0.8818
توقف زودهنگام	10	0.90904	0.8950

(سیر اجرای همه بخش‌های این قسمت در نوتبوک `ablation_study` آمده است.)

```yaml
batch_size: 8
early_stopping_threshold: 0
epochs: 50
layers_sizes:
- 64
learning_rate: 1e-3
logging_enable_print: false
logging_step: 1
loss_function: cross_entropy
momentum: 0.9
normalize: true
regularization_factor: 0.001
```

نمونه کد ۳: تنظیمات مربوط به بهترین مدل

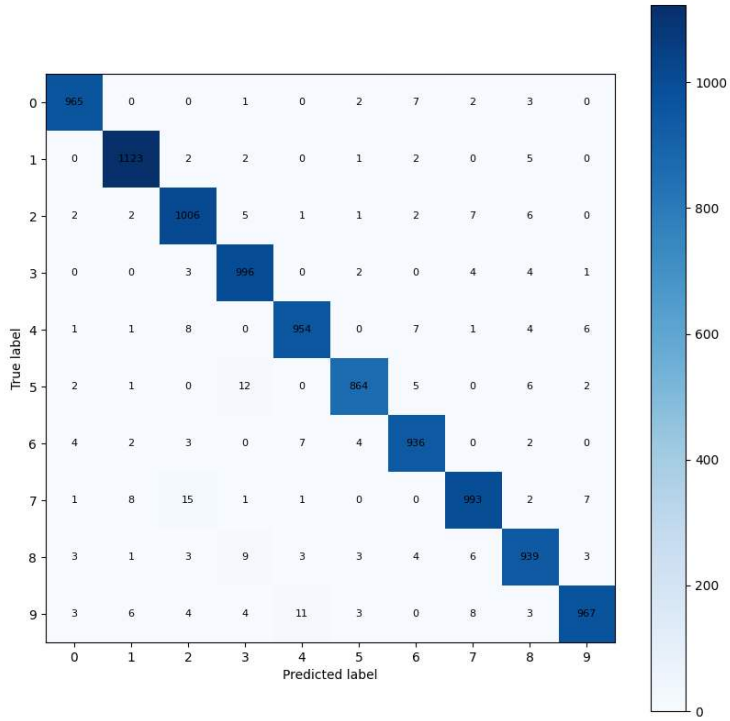
۱۵.۲ صحت و ماتریس درهم‌ریختگی

دقت مدل روی مجموعه‌داده‌های مختلف در جدول ۸.۲ آمده است.

ماتریس درهم‌ریختگی آن در شکل ۱۶.۲ آمده است.

جدول ۸.۲: دقت مدل نهایی روی مجموعه‌های آموزش، اعتبارسنجی و آزمون

آموزش	اعتبارسنجی	آزمون
0.99124	0.975	0.9743



شکل ۱۶.۲: ماتریس درهم‌ریختگی مدل نهایی برای مجموعه داده آزمون.

۱۶.۲ صحت مدل به ازای هر کلاس

در جدول ۹.۲ مقدار دقت کلاسی [۱۳] ، یادآوری [۱۴] و امتیاز F_1 در هر کلاس آورده شده است.

[13] Precision
[14] Recall

جدول ۹.۲: دقت، یادآوری و امتیاز F_1 برای هر کلاس

کلاس	دقت کلاسی	یادآوری	F_1
0	0.98369	0.984694	0.984192
1	0.981643	0.989427	0.985520
2	0.963602	0.974806	0.969171
3	0.966990	0.986139	0.976471
4	0.976459	0.971487	0.973966
5	0.981818	0.968610	0.975169
6	0.971963	0.977035	0.974492
7	0.972576	0.965953	0.969253
8	0.964066	0.964066	0.964066
9	0.980730	0.958375	0.969424

جهت تحلیل راحت‌تر، کلاس‌ها براساس دقت کلاسی و یادآوری مرتب شدند و در جدول ۱۰.۲ به نمایش درآمده‌اند. طبق تعریف، کلاس‌هایی که دقت کلاسی پایینی دارند، یعنی مدل سایر کلاس‌ها را به اشتباه، آن کلاس درنظر می‌گیرد و کلاس‌هایی که یادآوری پایینی دارند یعنی مدل نتوانسته داده با برچسب آن کلاس را درست طبقه‌بندی کند.

می‌توان دید که پایین‌ترین دقت کلاسی را کلاس‌هایی مثل ۲، ۸ و ۳ دارند. احتمالا چون سایر ارقام می‌توانند شبیه آن‌ها نوشته شوند، مدل به اشتباه این کلاس را برای آن‌ها درنظر گرفته است. برای یادآوری نیز، می‌توان دید ارقامی مثل ۹ و ۷ یادآوری پایینی دارند. این مورد نیز می‌تواند به علت شباهت دو کلاس (برای مثال اگر حلقه پایین ۹ نوشته نشود، شبیه به ۴ می‌شود.) و یا این باشد که این ارقام می‌توانند به شیوه‌های گوناگونی نوشته شوند که بعضا خواندن و تشخیص آن‌ها حتی برای انسان هم سخت باشد و از این رو مدل به اشتباه افتاده است.

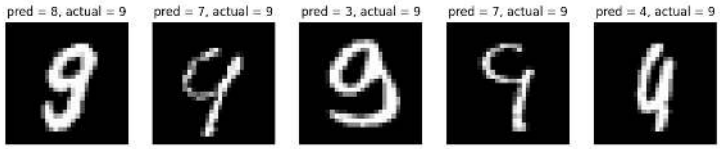
جدول ۱۰.۲: رتبه‌بندی کلاس‌ها بر اساس دقت کلاسی و یادآوری (صعودی)

رتبه	دقت کلاسی	یادآوری
1	2	9
2	8	8
3	3	7
4	6	5
5	7	4
6	4	2
7	9	6
8	1	0
9	5	3
10	0	1

۱۷.۲ کلاس با بیشترین اشتباه

سوال عملا کلاسی را می‌خواهد که پایین‌ترین یادآوری را داشته باشد که کلاس ۹ می‌شود. در بخش قبل مختصر توضیح داده شد. اگر به ماتریس درهم‌ریختگی در شکل ۱۶.۲ نگاه می‌کنیم، می‌توان دید که کلاس ۹ عمدتا با ۴ (۱۱ مورد) و ۷ (۸ مورد) اشتباه گرفته شده که این به علت این است که از آنجایی که ۹ را می‌توان به شیوه‌های مختلفی نوشت، در حالتی که حلقه پایین آن نوشته نشود شبیه به ۴ می‌شود و در حالتی که حلقه چپ آن کامل بسته نشود شبیه به ۷ می‌شود.

همچنین چون از منظم‌سازی استفاده شده، باعث می‌شود تا مدل به یک قسمت خاص حساس نشود. این کار باعث می‌شود تا یک نویز خاص در یک گوشه تصویر و یا مثلا یک دم اضافه در نوشتار منجر به تشخیص اشتباه نشود ولی ممکن است یادآوری را پایین آورد زیرا برای مثال در تشخیص رقم ۹، مدل باید به حلقه پایین حساس باشد که بتواند کلاس ۹ را از کلاس ۴ جدا کند ولی منظم‌سازی جلوی حساسیت بالا را می‌گیرد.

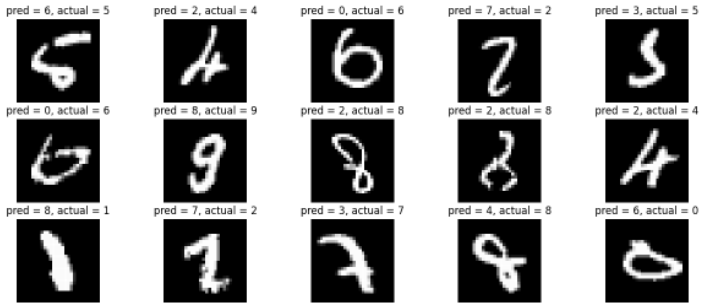


شکل ۱۷.۲: نمونه تصاویر رقم ۹ که به اشتباه دسته‌بندی شده‌اند.

۱۸.۲ کلاس‌هایی که با یکدیگر اشتباه گرفته می‌شوند

عملا باید دید کدام خانه غیرسطری در ماتریس درهم‌ریختگی بیشترین مقدار را دارد. می‌توان دید که زوج‌های (۷، ۲)، (۵، ۳) و (۹، ۴) بیشترین مقدار را دارند. هر سه این زوج‌ها نوشتار مشابهی دارند و همین مشابهت باعث شده تا مدل در تشخیص آن‌ها دچار اشتباه شود.

۱۹.۲ نمونه‌های اشتباه



شکل ۱۸.۲: نمونه تصاویری که اشتباه دسته‌بندی شده‌اند.

در شکل ۱۸.۲ چند مورد از مواردی که به اشتباه دسته‌بندی شده‌اند به نمایش درآورده شده‌اند. می‌توان دید مواردی از آن‌ها، مثلا عکس موجود در سطر اول و ستون دوم به دلیل ابهام خود تصویر به اشتباه دسته‌بندی شده‌اند. و یا موردی مثل عکس سطر اول و ستون پنجم اصلا قابل تشخیص نیست.

در مواردی مثل عکس ستون‌های چهارم و پنجم سطر دوم، عکس برای انسان قابل تشخیص است. اما تشخیص آن برای مدل نیاز به توجه زیاد به یک قسمت خاص از عکس دارد. ولی چون مدل فهم مکانی از تصویر ندارد (زیرا از لایه‌های تمام متصل [۱۵] استفاده شده است.) و همچنین چون منظم‌سازی انجام شده و سعی شده تا از دادن وزن بالا به یک سری پیکسل (و یا ویژگی) خاص جلوگیری شود، مدل نتوانسته آن‌ها را تشخیص دهد.

[15] Fully Connected