

به نام خدا



درس: یادگیری عمیق

استاد: دکتر فاطمی زاده

گزارش پروژه نهایی درس

سید محمد امین منصوری طهرانی

۹۴۱۰۵۱۷۴

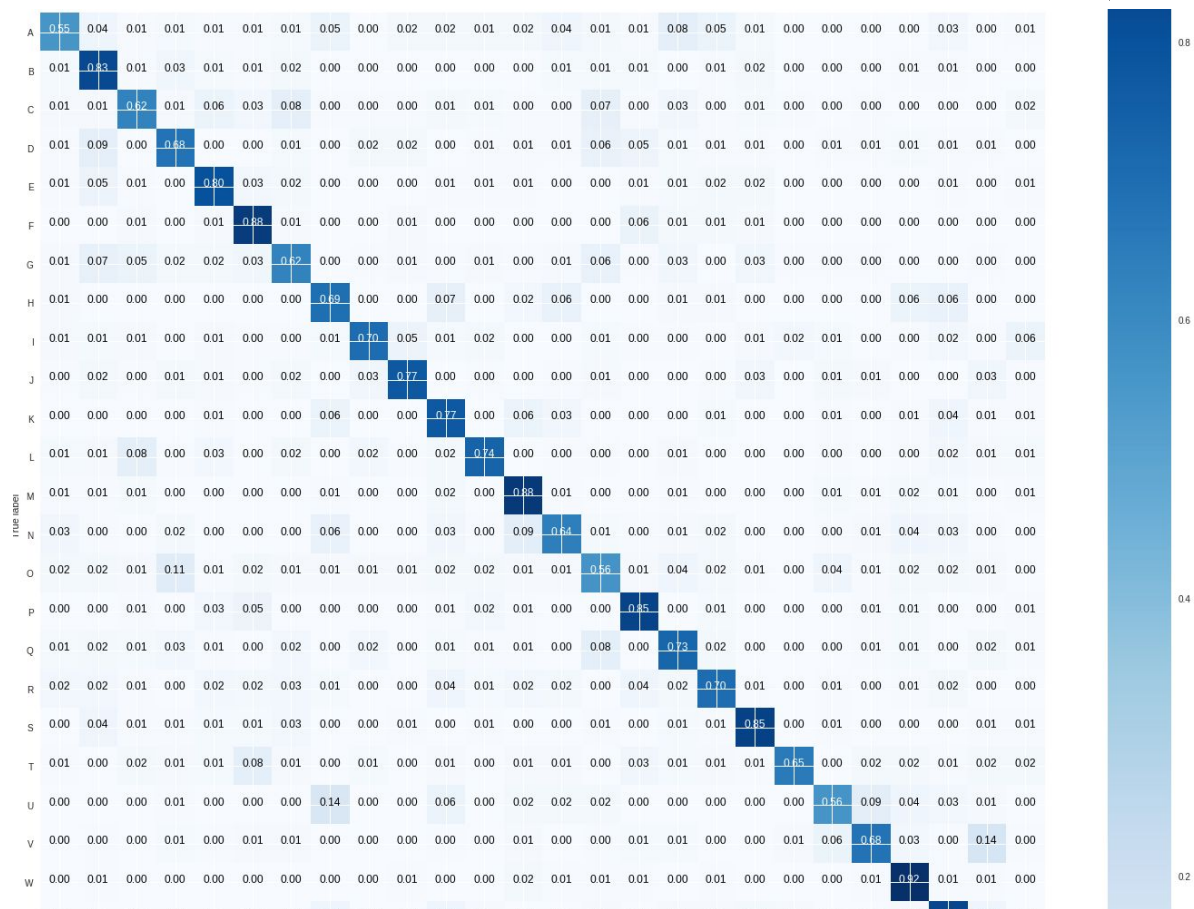
قسمت اول: طراحی دسته‌بند

کد این قسمت با نام Classifier پیوست شده‌است. (Jupyter Notebook) از آنجایی که پرداختن به جزئیات کدنویسی در تمرین‌ها انجام شد در این گزارش از اشاره به آن‌ها خودداری می‌کنیم. برای سادگی کار با کولب داده‌ها را پیکل کردم و در درایو گوگل از آن‌ها استفاده کردم و به این منظور برچسب‌ها نیز وان‌هات شدند. شبکه‌های مختلفی را برای بدست آوردن نتیجه مطلوب چک کردم. از جمله شبکه‌های سبکی که امتحان کردم استفاده از ۴ لایه کانولوشنی به همراه ۲ پولینگ و هر دفعه تعداد فیچر مپ‌ها را ۲ برابر کردم. بعد از این لایه‌های کانولوشنی با توجه به نیاز به feature extraction مناسب از ۳ لایه fc با تعداد نرون به ترتیب ۲۰۴۸ و ۵۱۲ و ۲۵۶ استفاده کردم. نتیجه این طبقه‌بند با ۲۰ اپیاک دقت مطلوبی نداد و نهایتاً به دقت ۷۳ درصد رسید.

```
[ ] [[160 10 1 8 5 1 16 13 4 7 10 14 12 13 10 1 6 25
      6 1 4 0 2 7 1 9]
      [ 1 128 2 0 5 2 6 1 0 0 0 5 0 0 2 1 1 4
      7 0 1 2 0 0 0 1]
      [ 0 0 146 0 7 1 2 0 0 0 0 0 0 0 0 1 0 0
      2 3 0 0 0 0 0 0]
      [ 1 19 2 148 1 3 3 3 1 1 0 1 4 6 19 13 3 2
      8 1 8 5 0 0 0 3]
      [ 0 1 9 0 130 1 2 0 0 0 0 1 1 0 0 0 0 0
      0 0 0 0 0 0 0 2]
      [ 0 1 1 1 6 150 0 0 0 0 0 0 0 0 2 0 0 0
      0 2 0 0 0 0 0 0]
      [ 0 0 7 0 1 1 134 0 0 0 0 0 0 0 3 0 5 0
      4 0 0 0 0 0 0 1]
      [ 2 0 0 0 1 0 0 146 0 0 5 1 8 10 1 1 1 1
      0 1 3 0 5 3 0 0]
      [ 2 1 1 1 4 3 2 0 128 3 4 4 0 2 2 2 3 1
      4 20 1 1 1 3 2 2]
      [ 0 3 1 5 0 4 5 2 50 179 0 1 3 1 3 0 2 1
      7 15 4 2 2 5 2 3]
      [ 0 0 0 0 1 0 0 4 0 0 130 0 0 0 0 0 0 1
      0 0 0 0 0 1 0 0]
      [ 0 2 5 0 1 0 1 0 4 1 1 164 0 0 2 0 0 0
      0 1 0 1 0 1 1 1]
      [ 1 2 0 0 1 0 0 0 0 0 0 0 132 2 0 0 0 0
      0 0 0 0 1 0 0 0]
      [ 2 3 0 0 1 0 0 4 0 0 5 1 29 147 2 1 0 2
      0 0 2 6 2 2 1 0]
      [ 3 3 7 14 3 4 10 1 4 4 2 3 2 4 157 4 11 0
      1 3 4 2 5 2 4 3]
      [ 1 0 1 3 1 24 0 0 0 0 0 0 0 1 0 144 0 3
      1 0 0 1 0 0 0 1]
      [ 4 3 4 2 4 0 13 2 0 1 1 2 5 1 16 2 163 5
      1 0 2 2 1 1 1 2]
      [ 1 2 1 0 2 3 0 0 0 0 0 0 1 0 1 0 3 0 129
      1 0 0 0 0 1 0 0]
      [ 1 4 1 2 7 1 5 0 5 6 1 0 1 1 3 1 2 2
      155 2 3 0 0 1 0 6]
      [ 0 0 3 0 1 7 0 0 2 0 1 0 0 0 0 3 0 0
      1 147 0 0 0 0 5 1]
      [ 0 1 0 1 0 0 1 0 0 0 3 1 1 1 1 1 2 0
      0 1 120 7 4 0 3 1]
      [ 0 1 0 1 0 0 2 0 0 0 0 0 2 0 0 1 0 0
      0 0 18 132 10 0 10 0]
      [ 0 1 0 0 0 0 1 0 0 1 1 4 3 0 0 0 3
      0 0 2 0 163 0 0 0]
      [ 0 0 0 0 1 0 0 4 2 0 29 0 4 2 1 0 0 1
      1 1 0 0 0 138 0 1]
      [ 2 2 0 3 1 2 3 3 1 1 2 1 3 7 1 8 0 1
```

The test accuracy is:0.7357692and its loss is:3.1836553

ماتریس درهم ریختگی آن:

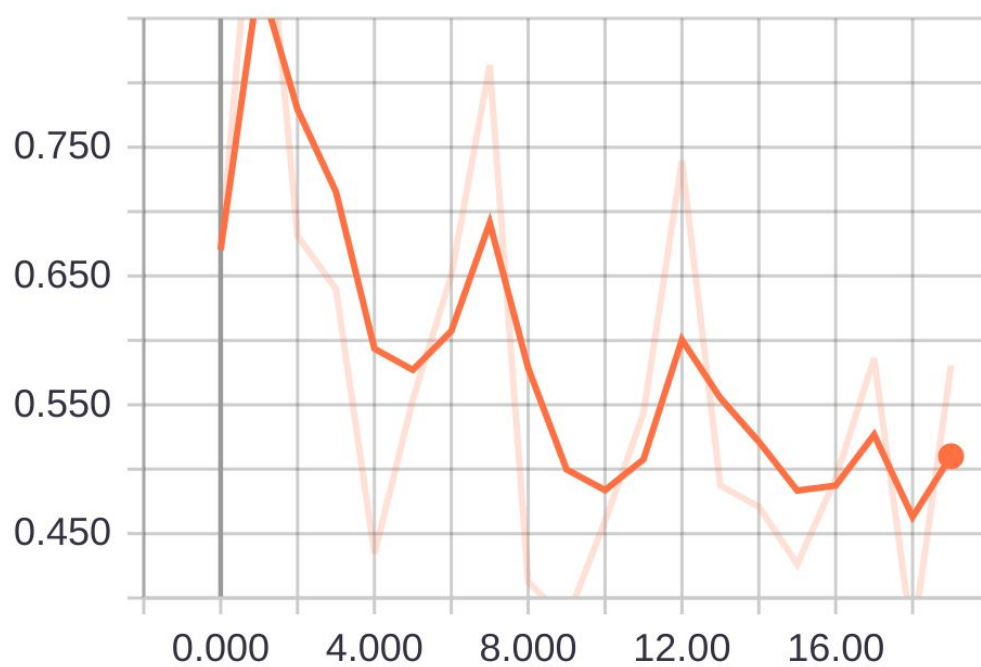


البته بهترین دقتی که به آن رسیدم ۸۱ درصد بود که متأسفانه از آن عکس نگرفتم. آن دقت با شبکه‌ای با ۶ لایه کانولوشنی و ۵ لایه تمام‌متصل حاصل شده بود که مدت زمان آموزش بسیار طولانی داشت و به همین علت به شبکه کم‌عمق تر فوق تغییرش دادم.

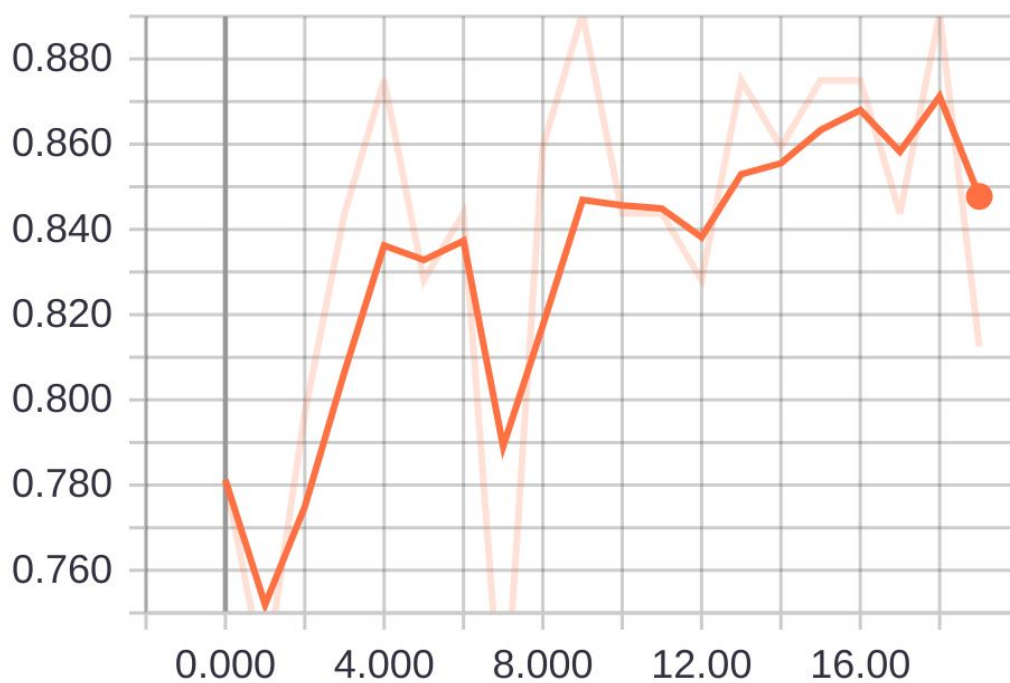
کد test1.ipynb نیز پیوست شده‌است.

نمودار تابع هزینه برای داده‌های تست و اعتبارسنجی:

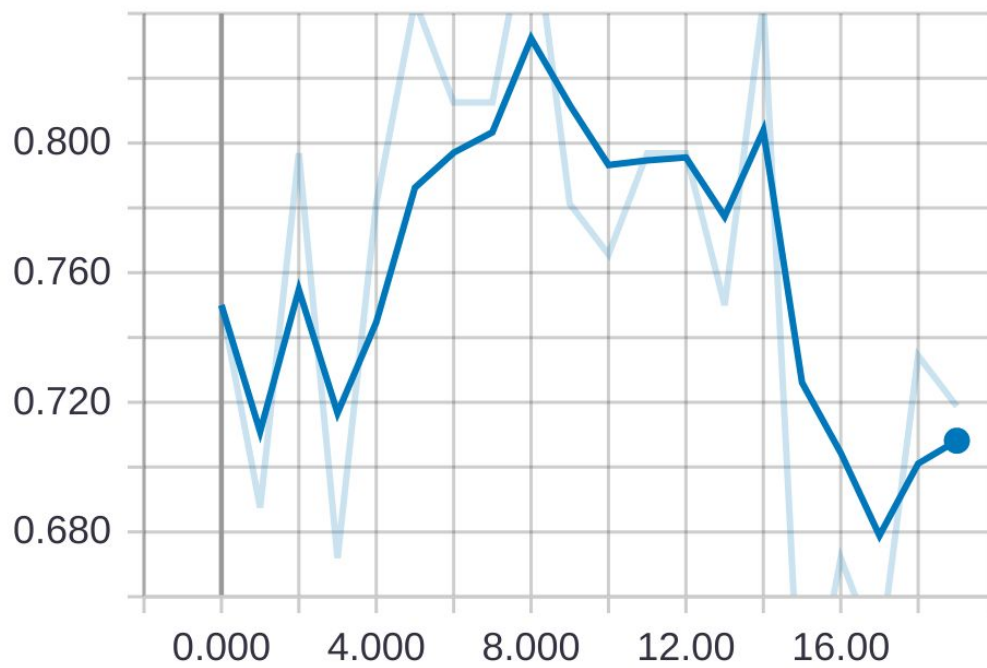
loss_train_1



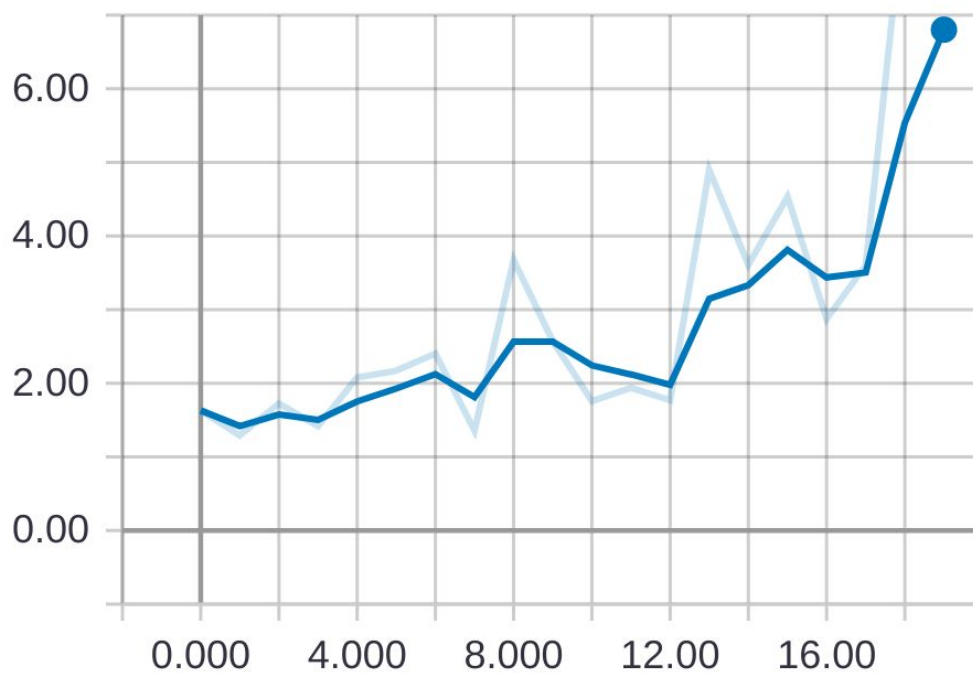
train_accuracy_1



validation_accuracy_1



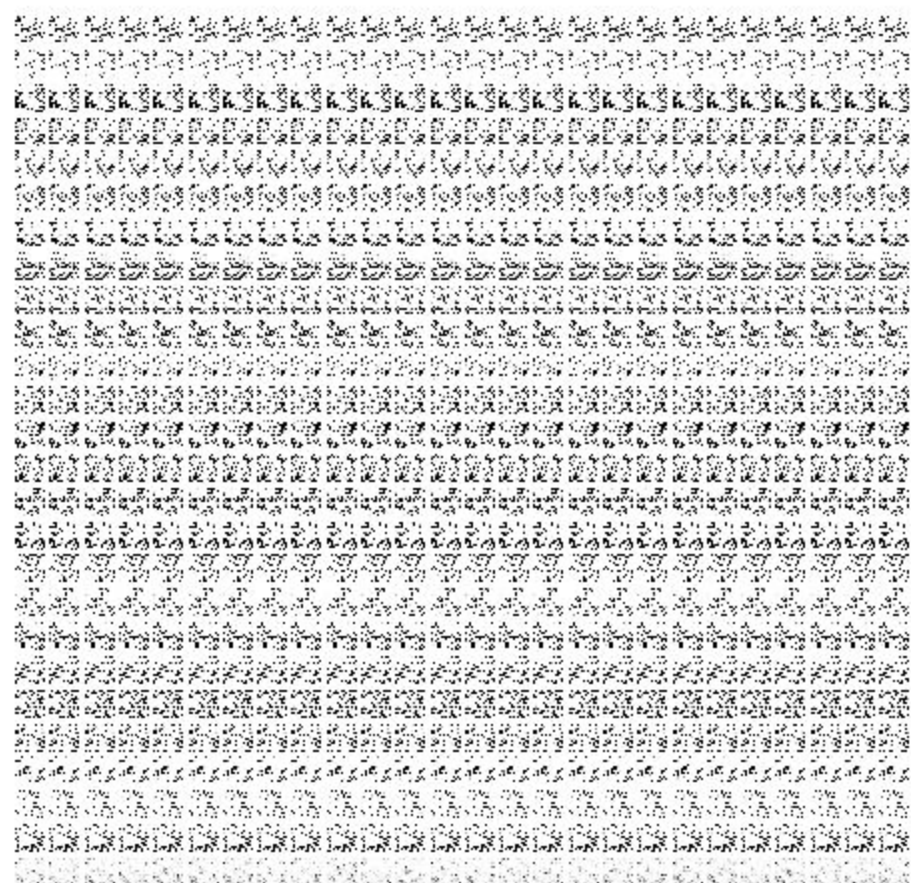
loss_validation_1



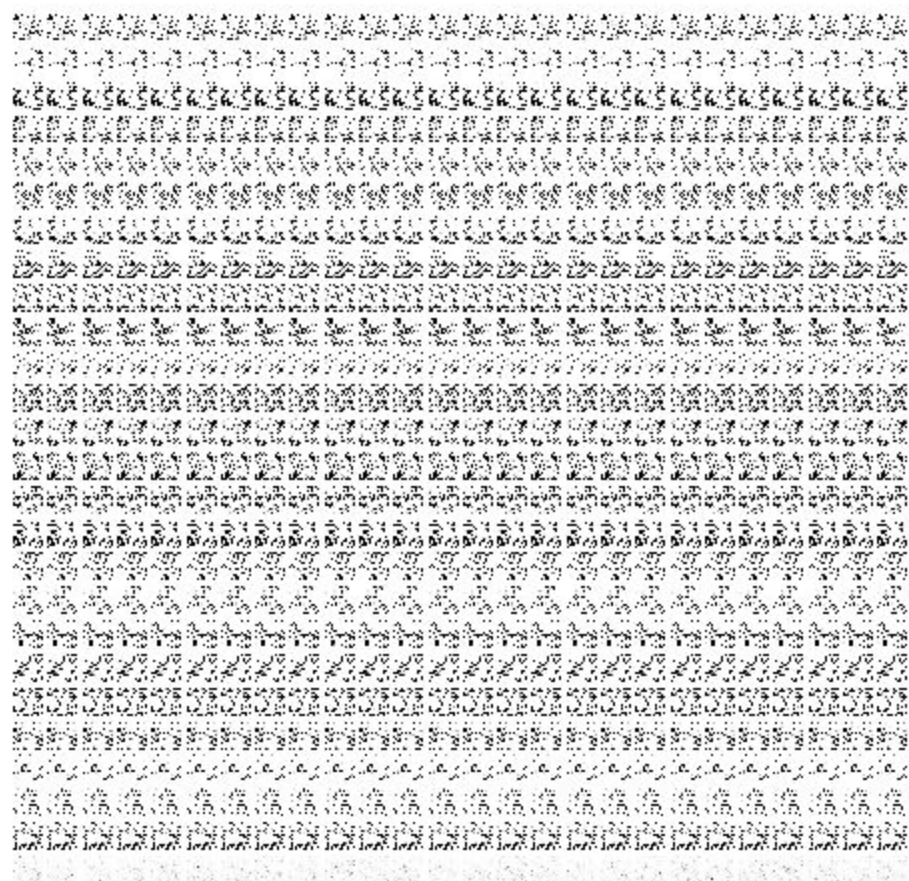
شبکه GAN:

برای این شبکه به وسیله ۳ لایه دی کانولوشن به تصویر ۶۴ در ۶۴ می‌رسیم و لیبل را به انتهای بردار latent می‌چسبانیم. برای تمایزدهنده نیز با ۳ لایه کانولوشنی که stride برابر ۲ دارند به یک تک عدد می‌رسیم. برای گذاشتن condition بر روی ورودی تمایز دهنده به این صورت عمل می‌کنیم که به صورت وان‌هات ۲۶ تصویر ۳۲ در ۳۲ پشت ورودی می‌چسبانیم و به تمایزدهنده می‌دهیم. در واقع اگر حرف A است، تصویر اول خود تصویر A و تصویر دوم یک تصویر ۳۲ در ۳۲ که همه درایه‌های آن یک باشند است. همه درایه‌های تصاویر بعدی صفر هستند. در هر مرحله آموزش نیز فقط یکی از تولید کننده و تمایزدهنده به روز می‌شوند.

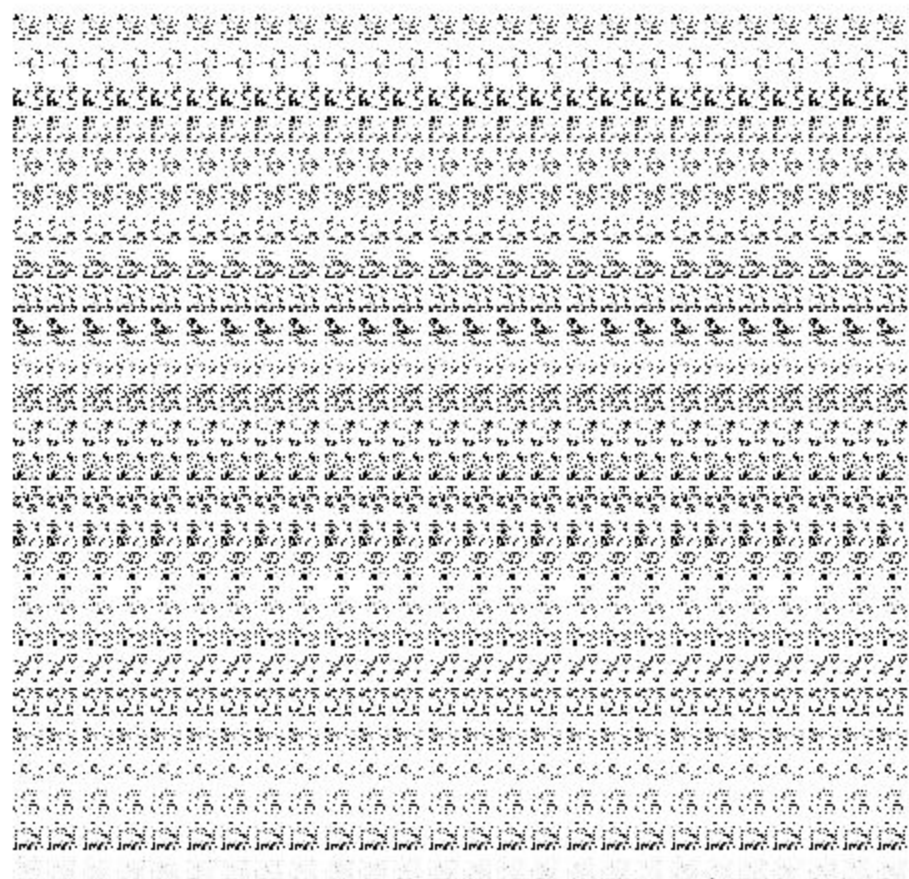
نتیجه شبکه به ازای ۱۰ اپیاک:



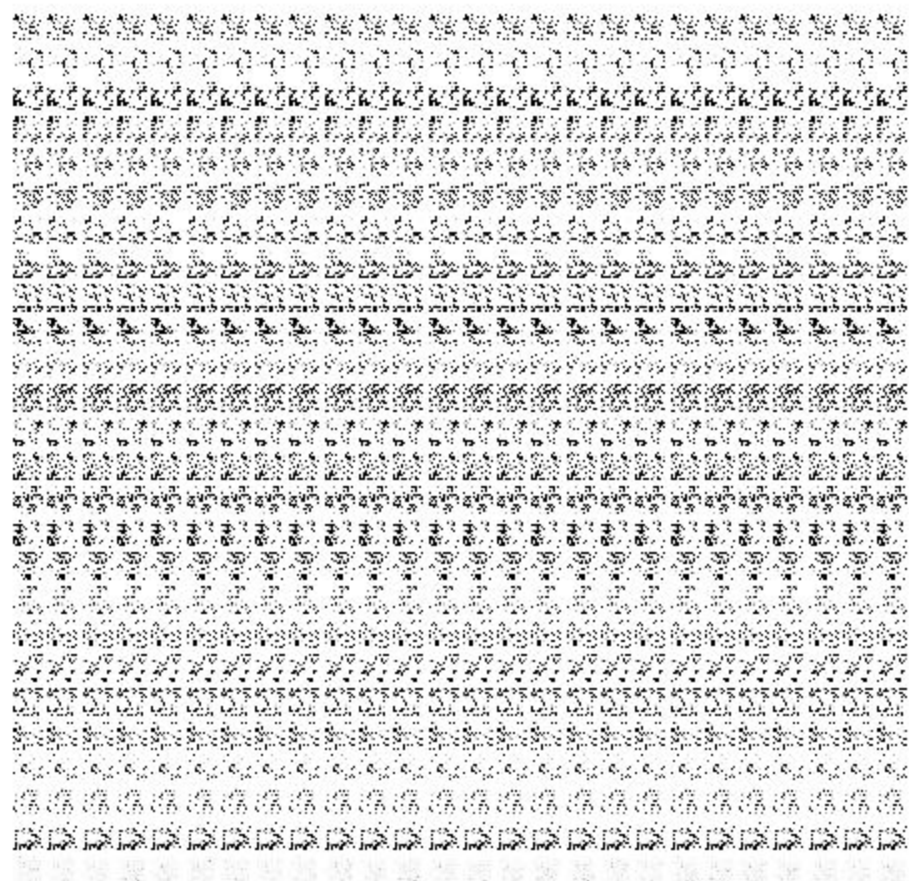
Epoch 1



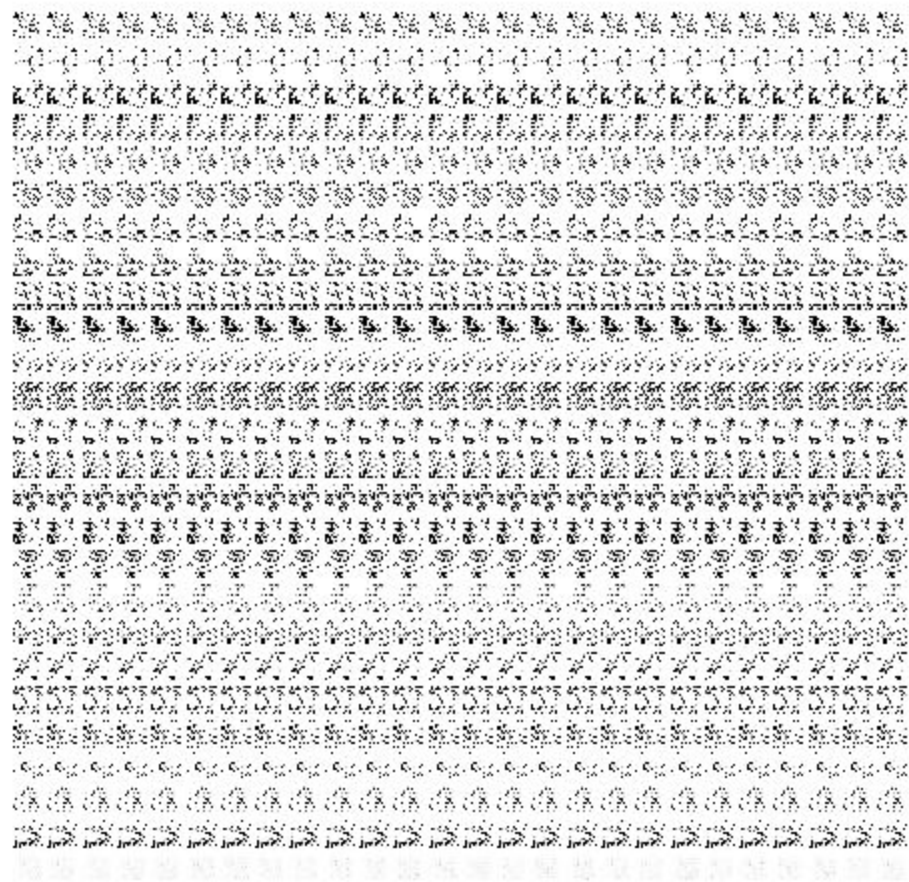
Epoch 2



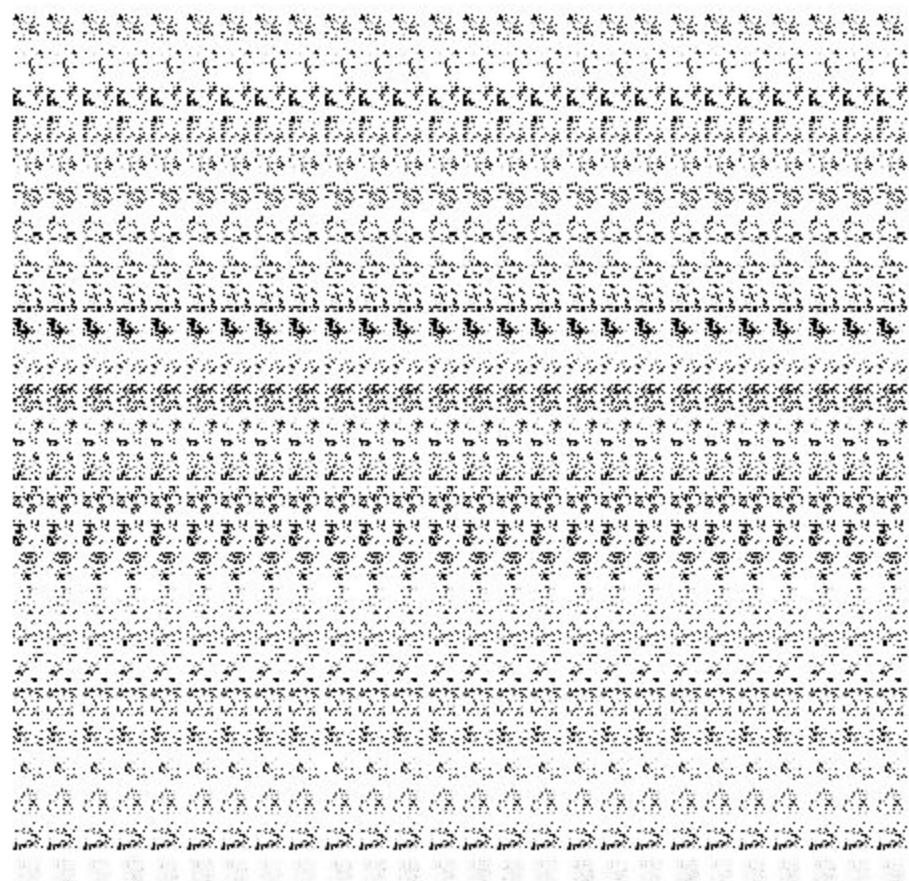
Epoch 3



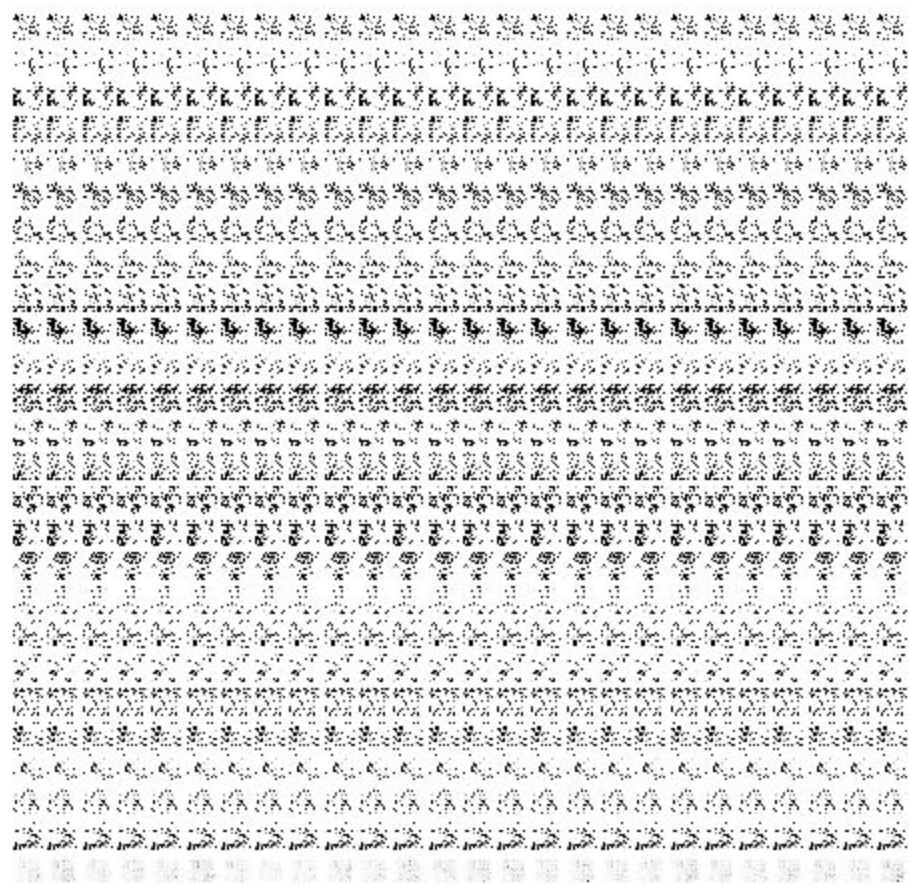
Epoch 4



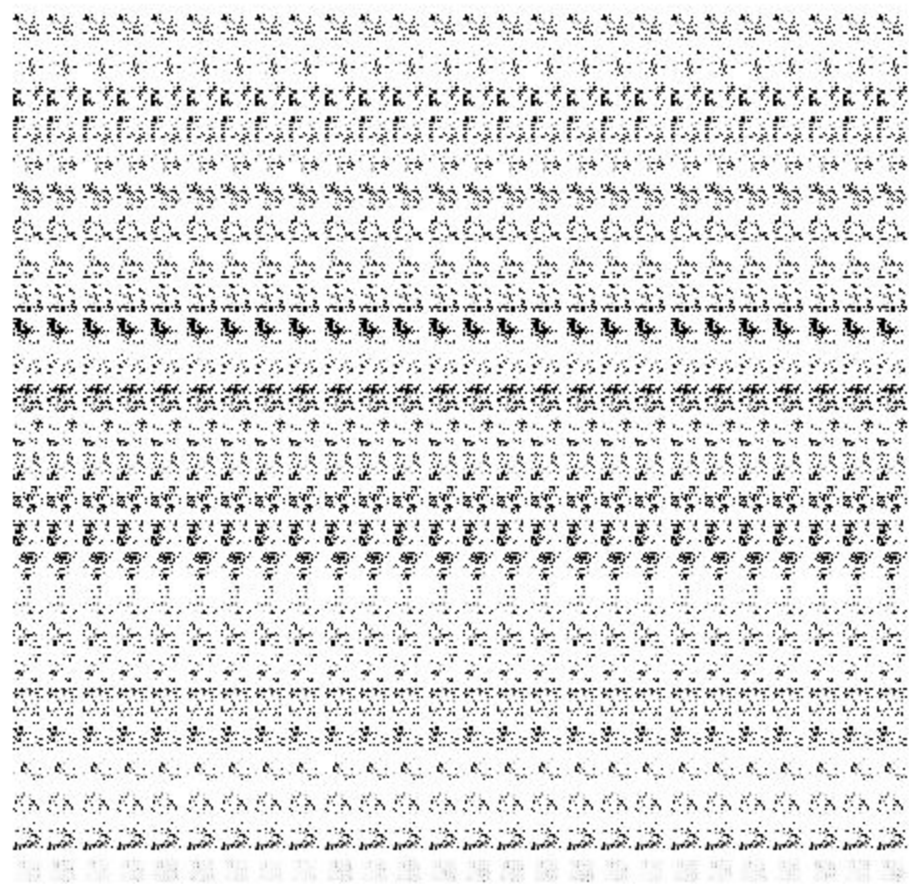
Epoch 5



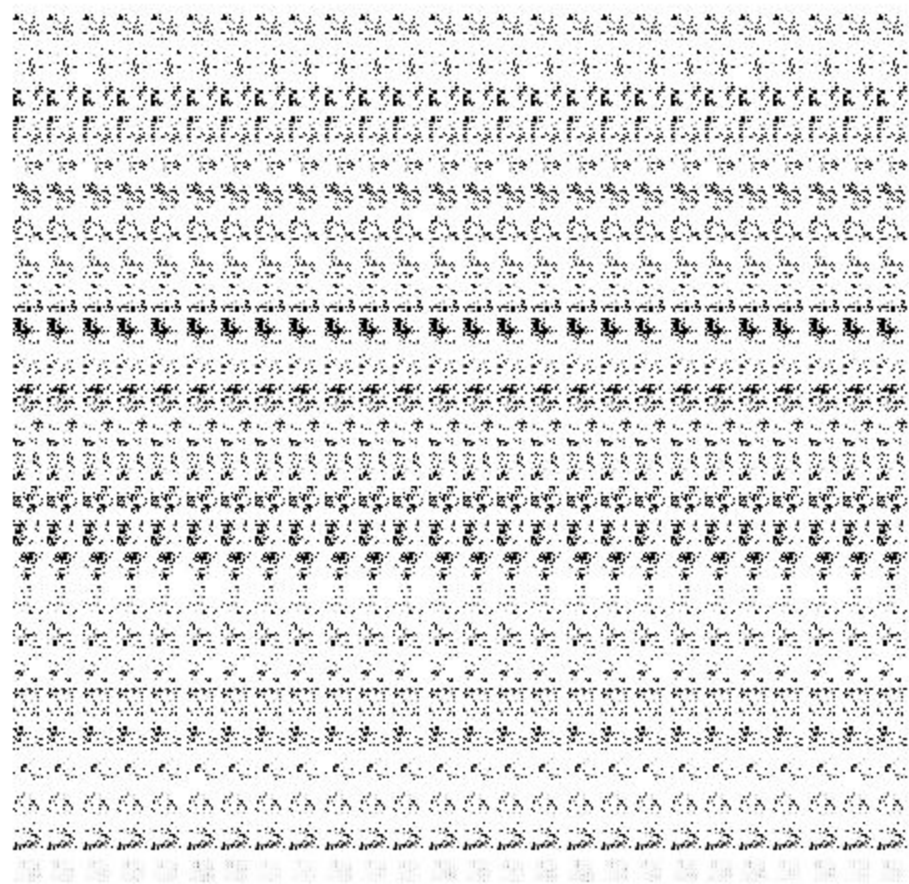
Epoch 6



Epoch 7



Epoch 8



Epoch 9