دانشکده مهندسی کامپیوتر

مباحث ویژه ۱ (یادگیری عمیق)

تمرین سری هفتم

علی صداقی

۹۷۵۲۱۳۷۸

۱ سوال اول

بایاس اختلاف میان میانگین پیش‌بینی‌های مدل ما نسبت به مقادیر صحیح است. خطای بایاس از مفروضات اشتباه در الگوریتم یادگیری ناشی می‌شود. به عبارت دیگر زمانی که دچار Underfit شده‌ایم.

واریانس میزان تغییرات پیش‌بینی مدل را نشان می‌دهد. خطای واریانس به دلیل حساسیت بالا نسبت به تغییرات کوچک در مجموعه داده ایجاد می‌شود. به عبارت دیگر زمانی که خوب Generalize نکردیم و دچار Overfit شده‌ایم.

- حالت Underfit: در این حالت شبکه روی داده آموزشی عملکرد مناسبی نداشته و مشکل High Bias را داریم. طبیعتا چون آموزش مناسب نبوده دقت روی داده ارزیابی هم نامناسب خواهد بود. درباره واریانس در حالت Underfit نمی‌توان نظر قطعی داد ولی در بیشتر اوقات Low Variance داریم زیرا شبکه چیز زیادی یاد نگرفته ولی همان چیزی را که یاد گرفته می‌تواند تعمیم دهد. در واقع در اکثر مدل‌هایی که دچار Underfit هستند، دقت فاز آموزش نزدیک دقت فاز ارزیابی است.
  راه‌حل‌ها:
  - افزایش پیچیدگی مدل
  - افزایش زمان آموزش (تعداد Epoch بیشتر)
  - انتخاب هایپرپارامترهای بهتر
  - افزایش تعداد ویژگی‌های (Features) داده ورودی

- حالت Overfit: در این حالت شبکه روی داده آموزشی بیش از حد Fit شده است و دقت بالایی روی این داده دارد. پس دارای Low Bias هستیم. اما شبکه نتوانسته Generalize کند و تعمیم بر روی داده ارزیابی مناسب نیست و دقت کمی روی داده ارزیابی و تست داریم. پس مشکل High Variance را داریم. در واقع در این حالت اختلاف دقت آموزش و ارزیابی زیاد است. یکی از عواملی که باعث می‌شود این مشکل ایجاد شود این است که هایپرپارامترهای شبکه را فقط با توجه به دقت آموزش Tune کردیم.
  راه‌حل کلی در این حالت استفاده از روش‌های منظم‌سازی (Regularization) است.
  راه‌حل‌ها:
  - جریمه اندازه پارامترها (منظم‌سازی L2 و L1)
  - داده‌افزایی (Data Augmentation)
  - افزودن نویز به داده ورودی (هم فیچر هم لیبل)

- کاهش پیچیدگی مدل
- استفاده از مکانیزم‌های Early Stopping در فاز آموزش
- استفاده از لایه Dropout
- تنظیم کردن هایپرپارامترها با توجه به داده Validation

۲ سوال دوم

- Forward pass:

$$Z_{h1} = i_1 w_1 + i_2 w_2$$

$$h_1 = ReLU(Z_{h1})$$

$$Z_{h2} = i_1 w_3 + i_2 w_4$$

$$h_2 = ReLU(Z_{h2})$$

$$z = h_1 w_5 + h_2 w_6$$

$$\hat{y} = a = \sigma(z) = \frac{1}{1 + e^{-z}}$$

$$L(a, y) = (y - a)^2 + \frac{\alpha}{2}(\sum_{i=1}^{6} w_i^2)$$

✓ We won't consider L2 effect in backpropagation. We consider it's effect on weight updating.

- Backward pass for single example:

$$\frac{dL(a,y)}{da} = -2 \times (y - a)$$

$$\frac{da}{dz} = \frac{-(-e^{-z})}{(1 + e^{-z})^2} = \frac{1}{1 + e^{-z}} \times \frac{e^{-z} + 1 - 1}{1 + e^{-z}} = \sigma(z) \times \big(1 - \sigma(z)\big) = a(1 - a)$$

$$\frac{dL}{dz} = \frac{dL}{da} \times \frac{da}{dz} = -2 \times (y - a)a(1 - a) = -2a(1 - a)(y - a)$$

$$\frac{dL}{dw_6} = \frac{dL}{dz} \times \frac{dz}{dw_6} = -2a(1 - a)(y - a) \times h_2$$

$$\frac{dL}{dw_5} = \frac{dL}{dz} \times \frac{dz}{dw_5} = -2a(1 - a)(y - a) \times h_1$$

$$\frac{dL}{dh_2} = \frac{dL}{dz} \times \frac{dz}{dh_2} = -2a(1 - a)(y - a) \times w_6$$

$$\frac{dh_2}{dw_4} = \frac{dh_2}{dZ_{h2}} \times \frac{dZ_{h2}}{dw_4} = (Z_{h2} > 0) \times i_2$$

$$\frac{dh_2}{dw_3} = \frac{dh_2}{dZ_{h2}} \times \frac{dZ_{h2}}{dw_3} = (Z_{h2} > 0) \times i_1$$

$$\frac{dL}{dw_4} = \frac{dL}{dh_2} \times \frac{dh_2}{dw_4} = -2a(1-a)(y-a) \times w_6 \times (Z_{h2} > 0) \times i_2$$

$$\frac{dL}{dw_3} = \frac{dL}{dh_2} \times \frac{dh_2}{dw_3} = -2a(1-a)(y-a) \times w_6 \times (Z_{h2} > 0) \times i_1$$

$$\frac{dL}{dh_1} = \frac{dL}{dz} \times \frac{dz}{dh_1} = -2a(1-a)(y-a) \times w_5$$

$$\frac{dh_1}{dw_2} = \frac{dh_1}{dZ_{h1}} \times \frac{dZ_{h1}}{dw_2} = (Z_{h1} > 0) \times i_2$$

$$\frac{dh_1}{dw_1} = \frac{dh_1}{dZ_{h1}} \times \frac{dZ_{h1}}{dw_1} = (Z_{h1} > 0) \times i_1$$

$$\frac{dL}{dw_2} = \frac{dL}{dh_1} \times \frac{dh_1}{dw_2} = -2a(1-a)(y-a) \times w_5 \times (Z_{h1} > 0) \times i_2$$

$$\frac{dL}{dw_1} = \frac{dL}{dh_1} \times \frac{dh_1}{dw_1} = -2a(1-a)(y-a) \times w_5 \times (Z_{h1} > 0) \times i_1$$

✓ For batch size 2 we need to take the average of calculated values.

- Update parameters with bias corrected Adam:

$$\nabla_{w_i}\bar{L} := \alpha w_i + \frac{dL}{dw_i}$$

$$v := \beta_1 \times v + (1-\beta_1) \times \nabla_{w_i}\bar{L}$$

$$s := \beta_2 \times s + (1-\beta_2) \times (\nabla_{w_i}\bar{L})^2$$

$$v := \frac{v}{1-\beta_1^t}$$

$$s := \frac{s}{1-\beta_2^t}$$

$$\Delta w_i = -\eta \frac{v}{\sqrt{s+\varepsilon}} \times \nabla_{w_i}\bar{L}$$

$$w_i := w_i + \Delta w_i$$

- Dataset:

N	X_n	Y_n
1	3, 2	0
2	15, 12	1

- Hyperparameters:

$$\alpha = 0.01$$

$$\beta_1 = 0.9$$

$$\beta_2 = 0.999$$

$$\varepsilon = 0$$

$$\eta = 0.01$$

- Initial weights:

$$w_6 = +0.5$$
$$w_5 = -1.0$$
$$w_4 = +1.5$$
$$w_3 = -0.5$$
$$w_2 = -2.5$$
$$w_1 = +2.0$$

محاسبات تا ۳ رقم اعشار

❖ Epoch 1:

- Data 1:

$$h_1 = ReLU(3 \times 2.0 + 2 \times -2.5) = ReLU(1) = 1$$

$$h_2 = ReLU(3 \times -0.5 + 2 \times 1.5) = ReLU(1.5) = 1.5$$

$$z = 1 \times -1.0 + 1.5 \times 0.5 = -0.25$$

$$a = \frac{1}{1 + e^{0.25}} = 0.437$$

$$L(a, y) = (y - a)^2 = (0 - 0.437)^2 = 0.190$$

$$\frac{dL}{dz} = -2(0.437)(1 - 0.437)(0 - 0.437) = 0.215$$

$$\frac{dL}{dw_6} = (0.215) \times (1.5) = 0.322$$

$$\frac{dL}{dw_5} = (0.215) \times (1) = 0.215$$

$$\frac{dL}{dw_4} = (0.215) \times (0.5) \times (1) \times (2) = 0.215$$

$$\frac{dL}{dw_3} = (0.215) \times (0.5) \times (1) \times (3) = 0.322$$

$$\frac{dL}{dw_2} = (0.215) \times (-1.0) \times (1) \times (2) = -0.430$$

$$\frac{dL}{dw_1} = (0.215) \times (-1.0) \times (1) \times (3) = 0.645$$

- L2 effect:

$$\nabla_{w_6} \bar{L} = (0.01) \times (0.5) + (0.322) = 0.327$$

$$\nabla_{w_5} \bar{L} = (0.01) \times (-1.0) + (0.215) = 0.205$$

$$\nabla_{w_4} \bar{L} = (0.01) \times (1.5) + (0.215) = 0.23$$

$$\nabla_{w_3} \bar{L} = (0.01) \times (-0.5) + (0.322) = 0.317$$

$$\nabla_{w_2} \bar{L} = (0.01) \times (-2.5) + (-0.430) = -0.455$$

$$\nabla_{w_1} \bar{L} = (0.01) \times (2.0) + (0.645) = 0.665$$

- Data 2:

$$h_1 = ReLU(15 \times 2.0 + 12 \times -2.5) = ReLU(0) = 0$$

$$h_2 = ReLU(15 \times -0.5 + 12 \times 1.5) = ReLU(10.5) = 10.5$$

$$z = 0.0 \times -1.0 + 10.5 \times 0.5 = 5.25$$

$$a = \frac{1}{1 + e^{-5.25}} = 0.994$$

$$L(a, y) = (y - a)^2 = (1 - 0.994)^2 = 0.000036$$

$$\frac{dL}{dz} = -2(0.994)(1 - 0.994)(1 - 0.994) = 0$$

$$\frac{dL}{dw_6} = (0) \times (10.5) = 0$$

$$\frac{dL}{dw_5} = (0) \times (0) = 0$$

$$\frac{dL}{dw_4} = (0) \times (0.5) \times (1) \times (12) = 0$$

$$\frac{dL}{dw_3} = (0) \times (0.5) \times (1) \times (15) = 0$$

$$\frac{dL}{dw_2} = (0) \times (-1.0) \times (0) \times (12) = 0$$

$$\frac{dL}{dw_1} = (0) \times (-1.0) \times (0) \times (15) = 0$$

- L2 effect:

$$\nabla_{w_6} \bar{L} = (0.327) \times (0.5) + (0) = 0.005$$

$$\nabla_{w_5} \bar{L} = (0.01) \times (-1.0) + (0) = -0.01$$

$$\nabla_{w_4} \bar{L} = (0.01) \times (1.5) + (0) = 0.015$$

$$\nabla_{w_3} \bar{L} = (0.01) \times (-0.5) + (0) = -0.005$$

$$\nabla_{w_2} \bar{L} = (0.01) \times (-2.5) + (0) = -0.025$$

$$\nabla_{w_1} \bar{L} = (0.01) \times (2.0) + (0) = 0.02$$

- Average:

$$L = 0.5 \times \big((0.190) + (0.000036)\big) = 0.095018$$

$$\nabla_{w_6} \bar{L} = 0.5 \times \big((0.327) + (0.005)\big) = 0.166$$

$$\nabla_{w_5} \bar{L} = 0.5 \times \big((0.205) + (-0.01)\big) = 0.097$$

$$\nabla_{w_4} \bar{L} = 0.5 \times \big((0.23) + (0.015)\big) = 0.122$$

$$\nabla_{w_3} \bar{L} = 0.5 \times \big((0.317) + (-0.005)\big) = 0.156$$

$$\nabla_{w_2} \bar{L} = 0.5 \times \big((-0.455) + (-0.025)\big) = -0.24$$

$$\nabla_{w_1} \bar{L} = 0.5 \times \big((0.665) + (0.02)\big) = 0.342$$

- Adam updating:

$$v_6 = (0.9) \times (0) + (1 - 0.9) \times (0.166) = 0.016 \xrightarrow{bias: \div 0.1} 0.166$$

$$v_5 = (0.9) \times (0) + (1 - 0.9) \times (0.097) = 0.009 \xrightarrow{bias: \div 0.1} 0.097$$

$$v_4 = (0.9) \times (0) + (1 - 0.9) \times (0.122) = 0.012 \xrightarrow{bias: \div 0.1} 0.122$$

$$v_3 = (0.9) \times (0) + (1 - 0.9) \times (0.156) = 0.015 \xrightarrow{bias: \div 0.1} 0.156$$

$$v_2 = (0.9) \times (0) + (1 - 0.9) \times (-0.24) = -0.024 \xrightarrow{bias: \div 0.1} -0.24$$

$$v_1 = (0.9) \times (0) + (1 - 0.9) \times (0.342) = 0.034 \xrightarrow{bias: \div 0.1} 0.342$$

$$s_6 = (0.999) \times (0) + (1 - 0.999) \times (0.166)^2 = 0.000027556 \xrightarrow{bias: \div 0.001} 0.027556$$

$$s_5 = (0.999) \times (0) + (1 - 0.999) \times (0.097)^2 = 0.000009409 \xrightarrow{bias: \div 0.001} 0.009409$$

$$s_4 = (0.999) \times (0) + (1 - 0.999) \times (0.122)^2 = 0.000014884 \xrightarrow{bias: \div 0.001} 0.014884$$

$$s_3 = (0.999) \times (0) + (1 - 0.999) \times (0.156)^2 = 0.000024336 \xrightarrow{bias: \div 0.001} 0.024336$$

$$s_2 = (0.999) \times (0) + (1 - 0.999) \times (-0.24)^2 = 0.0000576 \xrightarrow{bias: \div 0.001} 0.0576$$

$$s_1 = (0.999) \times (0) + (1 - 0.999) \times (0.342)^2 = 0.000116964 \xrightarrow{bias: \div 0.001} 0.116964$$

$$\Delta w_6 = (-0.01) \times \frac{(0.166)}{\sqrt{(0.027556)}} \times (0.166) = -0.00166$$

$$\Delta w_5 = (-0.01) \times \frac{(0.097)}{\sqrt{(0.009409)}} \times (0.097) = -0.00097$$

$$\Delta w_4 = (-0.01) \times \frac{(0.122)}{\sqrt{(0.014884)}} \times (0.122) = -0.00122$$

$$\Delta w_3 = (-0.01) \times \frac{(0.156)}{\sqrt{(0.024336)}} \times (0.156) = -0.00156$$

$$\Delta w_2 = (-0.01) \times \frac{(-0.24)}{\sqrt{(0.0576)}} \times (-0.24) = -0.0024$$

$$\Delta w_1 = (-0.01) \times \frac{(0.342)}{\sqrt{(0.116964)}} \times (0.342) = -0.00342$$

$$w_6 = (0.5) + (-0.00166) = 0.49834$$

$$w_5 = (-1.0) + (-0.00097) = -1.00097$$

$$w_4 = (1.5) + (-0.00122) = 1.49878$$

$$w_3 = (-0.5) + (-0.00156) = -0.50156$$

$$w_2 = (-2.5) + (-0.0024) = -2.5024$$

$$w_1 = (+2.0) + (-0.00342) = 1.99658$$

✓ We skipped similar computations and just reported important results.

❖ Epoch 2:

- Data 1:

$$h_1 = ReLU(3 \times 1.99658 + 2 \times -2.5024) = ReLU(0.98494) = 0.98494$$

$$h_2 = ReLU(3 \times -0.50156 + 2 \times 1.49878) = ReLU(1.5) = 1.49288$$

$$z = 0.98494 \times -1.00097 + 1.49288 \times 0.49834 = -0.2419335726$$

$$a = \frac{1}{1 + e^{0.2419335726}} = 0.43$$

$$L(a, y) = (y - a)^2 = (0 - 0.43)^2 = 0.1849$$

- Data 2:

$$h_1 = ReLU(15 \times 1.99658 + 12 \times -2.5024) = ReLU(-0.0801) = 0$$

$$h_2 = ReLU(15 \times -0.50156 + 12 \times 1.49878) = ReLU(10.46196) = 10.46196$$

$$z = 0.0 \times -1.00097 + 10.46196 \times 0.49834 = 5.2136131464$$

$$a = \frac{1}{1 + e^{-5.2136131464}} = 0.995$$

$$L(a, y) = (y - a)^2 = (1 - 0.995)^2 = 0.000025$$

- Average:

$$L = 0.5 \times \big((0.1849) + (0.000025)\big) = 0.0924625$$

- Adam updating:

$$\Delta w_6 = -0.001086$$

$$\Delta w_5 = -0.00069$$

$$\Delta w_4 = +0.0001$$

$$\Delta w_3 = -0.00107$$

$$\Delta w_2 = -0.0017$$

$$\Delta w_1 = -0.00235$$

$$w_6 = (0.49834) + (-0.001086) = 0.497254$$
$$w_5 = (-1.00097) + (-0.00069) = -1.00166$$
$$w_4 = (1.49878) + (+0.0001) = 1.49888$$
$$w_3 = (-0.50156) + (-0.00107) = -0.50263$$
$$w_2 = (-2.5024) + (-0.0017) = -2.5041$$
$$w_1 = (1.99658) + (-0.00235) = 1.99423$$

❖ Evaluation:

$$a_1 = 0.425$$
$$a_2 = 0.996$$
$$\bar{L} = 0.0917936$$

تفسیر:

همانطور که در بخش Epoch 1 Data 2 مشاهده شد ترکیب تابع فعال‌سازی Sigmoid و تابع ضرر MSE اصلا انتخاب مناسبی نیست زیرا در آن داده گرادیان به صفر میل کرد و نتوانستیم با آن داده وزن‌ها را آپدیت کنیم.

همچنین در هر ایپاک در یکی از خروجی‌های ReLU به مقدار صفر رسیدیم و باعث کمی مشکل Dying ReLU پیش بیاید. اما اثر آن طوری نبود که شبکه کلا دچار یخزدگی شود.

بررسی همگرایی:

$$y_1 = 0, \qquad a_1 = 0.437 \rightarrow 0.430 \rightarrow 0.425$$
$$y_2 = 1, \qquad a_2 = 0.994 \rightarrow 0.995 \rightarrow 0.996$$

وزن‌های اولیه طوری بودند که برروی داده دوم دقت خوبی داشتیم و تغییرات در جهت آن داده کمتر از داده اول بود.

$$\bar{L} = 0.095018 \rightarrow 0.0924625 \rightarrow 0.0917936$$

میانگین خطای MSE برای دو داده در حال کم شدن است اما سرعت کم شدن آن بسیار کم است و شیب منحنی خطا در این نقطه نزدیک ۰ است. اما همچنان در حال کاهش است زیرا روی داده اول خطای زیادی داریم. با توجه به کم شدن این خطا می‌توانیم بگوییم در جهت همگرایی در حرکت هستیم.

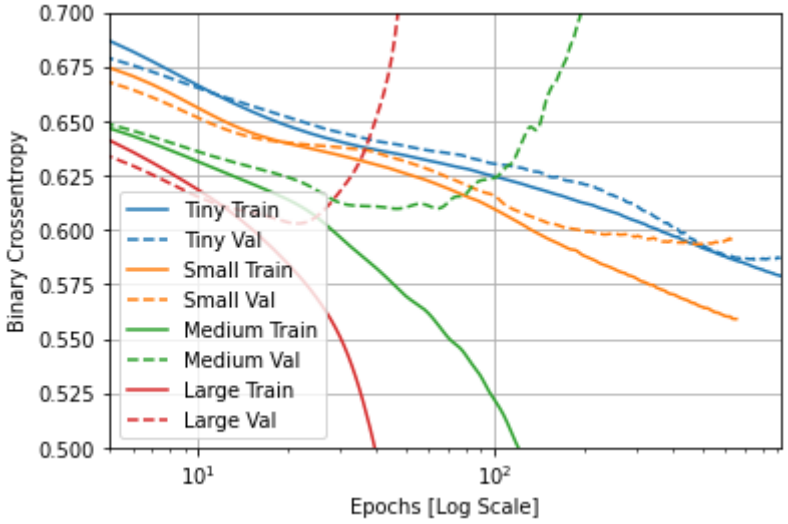
با توجه به اینکه تعداد تکرار در این الگوریتم برابر ۲ بود و همچنین از Batch Gradient Descent استفاده کردیم می‌توان گفت داده دوم با خطای صفر قدرت عمل داده اول با خطای بالا را نصف می‌کند و نمی‌گذارد خطا با سرعت بیشتری کاهش یابد. با توجه به کم بودن داده‌ها و همچنین کم بودن تعداد تکرار بهتر بود از حالت Stochastic Gradient Descent استفاده کنیم تا داده اول با قدرت بیشتری مارا به سمت همگرایی پیش ببرد.

بهینه‌ساز آدام با تنظیم کردن طول قدم در جهت هر وزن باعث شده سرعت بهینه‌سازی در جهات مختلف متفاوت باشد و همگرایی سریع‌تر رخ بدهد. همچنین با تصحیح بایاس در این بهینه‌ساز طول قدم در ایپاک‌های اولیه را بیشتر کردیم تا سرعت بیشتر شود.

منظم‌ساز *L2* در این سوال به‌نوعی در نقش *Bottle Neck* (گلوگاه) عمل کرده است. زیرا سرعت بهینه-سازی را کاهش داده در نتیجه حرکت به نقطه همگرایی کندتر شده. این اتفاق در حالتی رخ داده است که داده‌ای برای ارزیابی نداریم که بگوییم این منظم‌ساز به قدرت تعمیم‌دهی (*Generalization*) الگوریتم را افزایش داده است.

۳ سوال سوم

الف) با توجه به اینکه صورت سوال قسمت Underfit را نخواسته از قسمت‌های که مربوط به آن بخش است و شامل بزرگ کردن مدل و افزایش ظرفیت آن است عبور می‌کنیم. به طور خلاصه در قسمت مربوط به Underfit همه‌ی مدل‌ها به جز مدل Tiny دچار Overfit شدند و هر چه مدل بزرگ‌تر می‌شد Overfit بیشتری پیدا می‌کرد. مدل Tiny نیز مشکل Underfit را داشت.



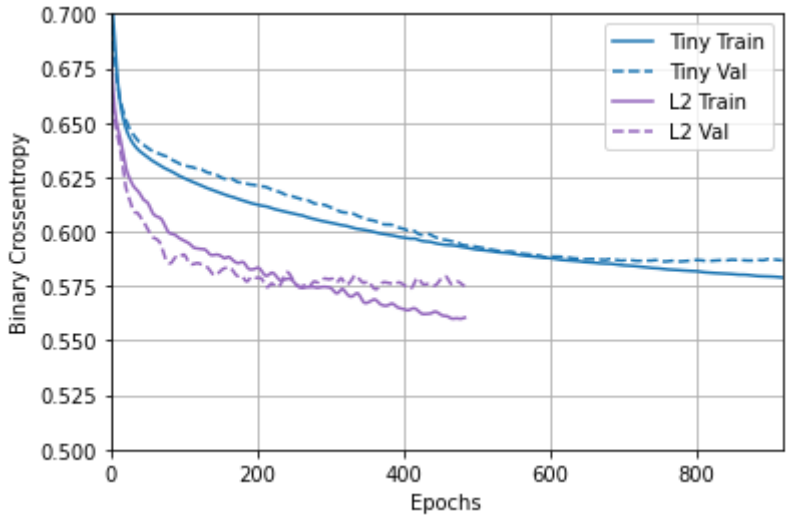
مدل استفاده شده در موارد زیر همان مدل Large قسمت Underfit می‌باشد که شامل 4 لایه Dense با 512 نورون و تابع فعال‌سازی elu است. در لایه آخر نیز یک لایه Dense با یک نورون بکار رفته است.

در تمامی حالات از Early Stopping نیز بهره برده شده است.

استراتژی‌ها برای جلوگیری از Overfit به صورت زیر بکار رفته:

- منظم‌سازی وزن‌ها Large + L2:

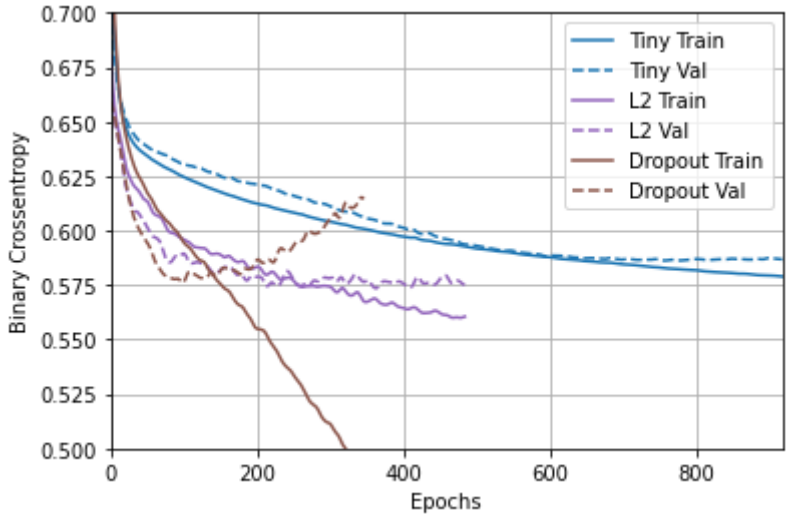
4 لایه Dense اولیه از kernel_regularizer حالت L2 استفاده کرده‌اند و مقدار alpha برابر 0.001 است.



همانطور که مشاهده می‌شود مدل نسبت به حالت Tiny که در قسمت اول دچار Underfit بود عملکرد بسیار بهتری داشته و دیگر دچار Overfit نشده است. عملکرد مدل نسبت به حالت پایه Large نیز بسیار بهتر شده و مشکل Overfit بسیار کمتر شده است. دقت در فاز آموزش حدود ۳ درصد بیشتر از فاز ارزیابی است.

- استفاده از Dropout یعنی Large + Dropout:

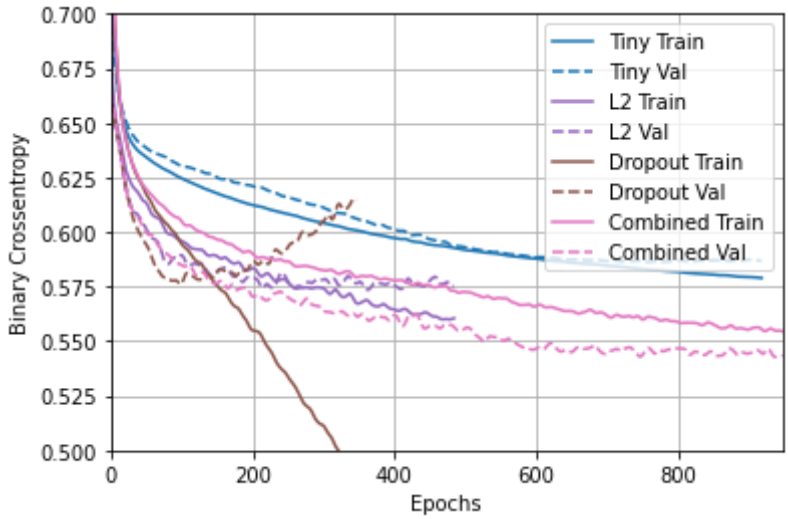
پس از هر 4 لایه Dense ابتدایی یک لایه Dropout با احتمال خاموشی 0.5 استفاده شده است.



همانطور که مشاهده می‌کنید مدل همچنان مشکل Overfit را دارد اما نسبت به حالت مدل Large عملکرد بهتری داشته. اما نسبت به حالت مدل Large + L2 عملکرد ضعیف تری دارد و Overfit بیشتری رخ داده است. دقت در فاز آموزش ۴ درصد بیشتر از فاز ارزیابی است.

- استفاده همزمان از L2 و Dropout یعنی Combined Large:

این حالت ترکیب دو روش بالا می‌باشد یعنی در هر 4 لایه ابتدایی Dense از kernel_regularizer حالت L2 با مقدار آلفای 0.0001 استفاده کردیم (اثر منظم‌سازی را کمتر کردیم). همچنین بعد از هر 4 لایه Dense یک لایه Dropout با احتمال 0.5 استفاده کردیم.



همانطور که مشاهده می‌شود عملکرد مدل خیلی بهتر شده و توانستیم بدون Overfit مدل را در مدت طولانی تری آموزش دهیم. این حالت نسبت به تمامی حالت دارای کمترین مقدار Overfit است و همچنین بهترین تعمیم و کمترین خطا را بر روی داده ارزیابی دارد. دقت در فاز آموزش تقریبا با فاز ارزیابی برابر است.

- ✓ استفاده از مدل کوچک‌تر، داده‌افزایی و Batch Normalization نیز برای کاهش Overfit بهتر است که در این مقاله به طور صریح بررسی نشده بودند.
- ✓ معمولا ترکیب روش‌های مختلف با هم باعث اثربخشی بهتر می‌شود.

ب) برای بررسی Underfit چهار مدل Simple، Small، Medium و Large را بدون هیچ منظم‌سازی در نظر می‌گیریم و تمامی مدل‌هارا در 1000 تکرار آموزش می‌دهیم. Early Stop را حذف کردیم و جای آن از Model Checkpoint استفاده کردیم تا بتوانیم شماره و وزن‌های بهترین Epoch را داشته باشیم.

```
Model: "Simple"
_________________________________________________________________
 Layer (type)                 Output Shape              Param #
=================================================================
 dense (Dense)               (None, 16)                464

 dense (Dense)               (None, 1)                 17
```

```
=================================================================
Total params: 481
Trainable params: 481
Non-trainable params: 0
```

Model: "Small"

```
_________________________________________________________________
 Layer (type)                 Output Shape              Param #
=================================================================
 dense (Dense)               (None, 32)                928

 dense (Dense)               (None, 1)                 33

=================================================================
Total params: 961
Trainable params: 961
Non-trainable params: 0
```

Model: "Medium"

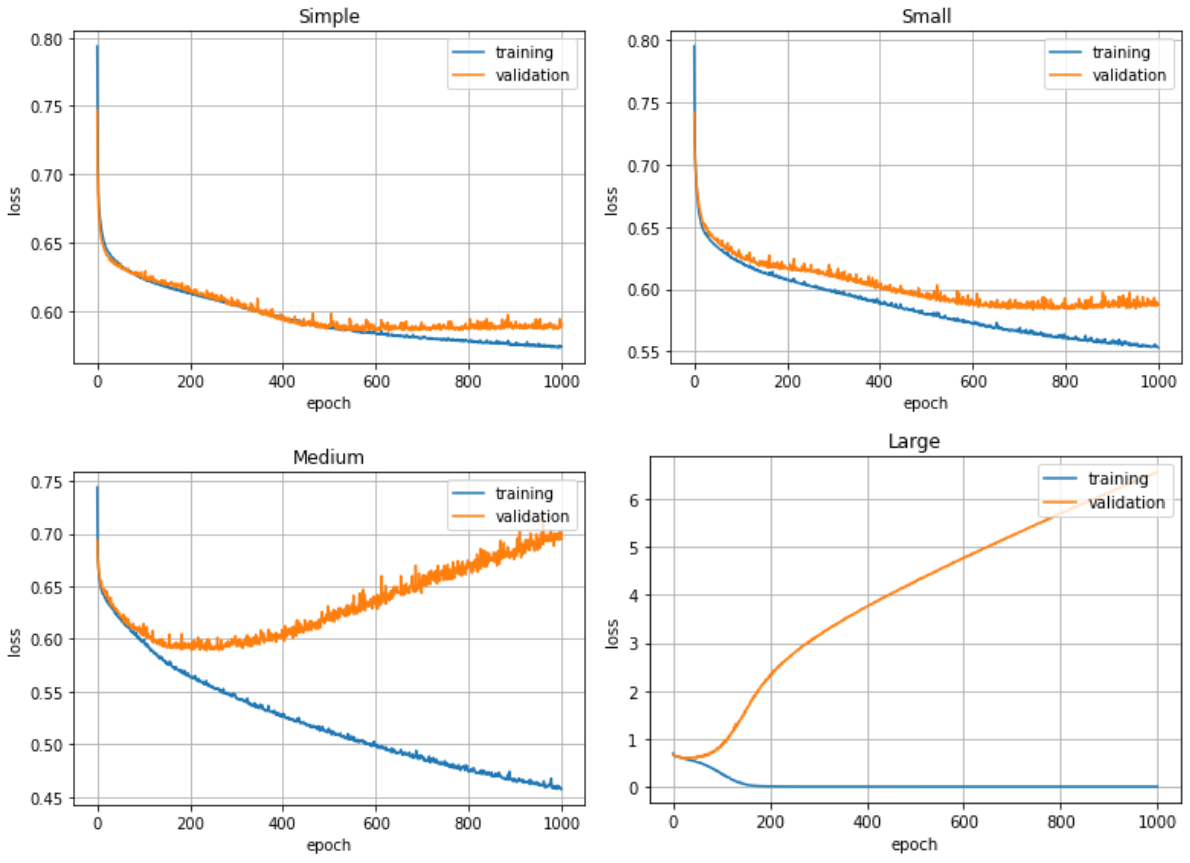
```
_________________________________________________________________
 Layer (type)                 Output Shape              Param #
=================================================================
 dense (Dense)               (None, 32)                928

 dense (Dense)               (None, 32)                1056

 dense (Dense)               (None, 1)                 33

=================================================================
Total params: 2,017
Trainable params: 2,017
Non-trainable params: 0
```

Model: "Large"

```
_________________________________________________________________
 Layer (type)                 Output Shape              Param #
=================================================================
 dense (Dense)               (None, 128)               3712

 dense (Dense)               (None, 128)               16512

 dense (Dense)               (None, 128)               16512

 dense (Dense)               (None, 128)               16512

 dense (Dense)               (None, 1)                 129

=================================================================
Total params: 53,377
Trainable params: 53,377
Non-trainable params: 0
```

نتایج در 1000 بار تکرار:

Model	Params	Best Epoch	Train Best	Test Best	Train Last	Test Last
Simple	481	926	loss: 0.5784 acc: 0.6974	loss: 0.5886 acc: 0.6980	loss: 0.5740 acc: 0.6821	loss: 0.5907 acc: 0.6680
Small	961	737	loss: 0.5672 acc: 0.7062	loss: 0.5859 acc: 0.6830	loss: 0.5531 acc: 0.6951	loss: 0.5879 acc: 0.6530
Medium	2,017	273	loss: 0.5476 acc: 0.7207	loss: 0.5908 acc: 0.7040	loss: 0.4575 acc: 0.7665	loss: 0.6948 acc: 0.6580
Large	53,377	42	loss: 0.5393 acc: 0.7275	loss: 0.6048 acc: 0.6860	loss: 0.0000 acc: 1.0000	loss: 6.5633 acc: 0.6340



برای بررسی Underfit باید به ستون Train Last و مقدار Accuracy نگاه کنیم. واضح است که مدل‌های Simple و Small بسیار Underfit هستند (دقت حدود ۶۹ درصد). وضع مدل Medium کمی بهتر است (دقت ۷۶ درصد) اما همچنان Underfit است. دقت در مدل Large برابر ۱۰۰ درصد شده‌است پس به هیچ عنوان دارای مشکل Underfit نیست.

برای بررسی مشکل Overfit باید به ستون Test Last و مقدار Accuracy توجه کنیم.

- مدل Simple تقریبا هیچ Overfitیی ندارد و دقت آموزش و ارزیابی با هم برابر ۶۹ درصد می‌باشد.
- مدل Small کمی دچار Overfit است و این اتفاق از ایپاک 737 رخ داده و منجر شده که دقت ارزیابی ۴ درصد کمتر از دقت آموزش است.
- مدل Medium مشکل Overfit بیشتری داشته و این اتفاق از ایپاک 273 رخ داده و منجر شده که دقت ارزیابی ۱۱ درصد کمتر از دقت آموزش است.
- مدل Large مشکل Overfit بسیار زیادی دارد و در همان ابتدا یعنی ایپاک 42 شروع به Overfit کرده. این مدل داده آموزش را به طور کامل حفظ شده و دقتش روی فاز آموزش ۱۰۰ درصد و خطا ۰ می‌باشد. اما بر روی داده ارزیابی عملکرد اصلا جالبی نداشته‌است. (۶۳ درصد).

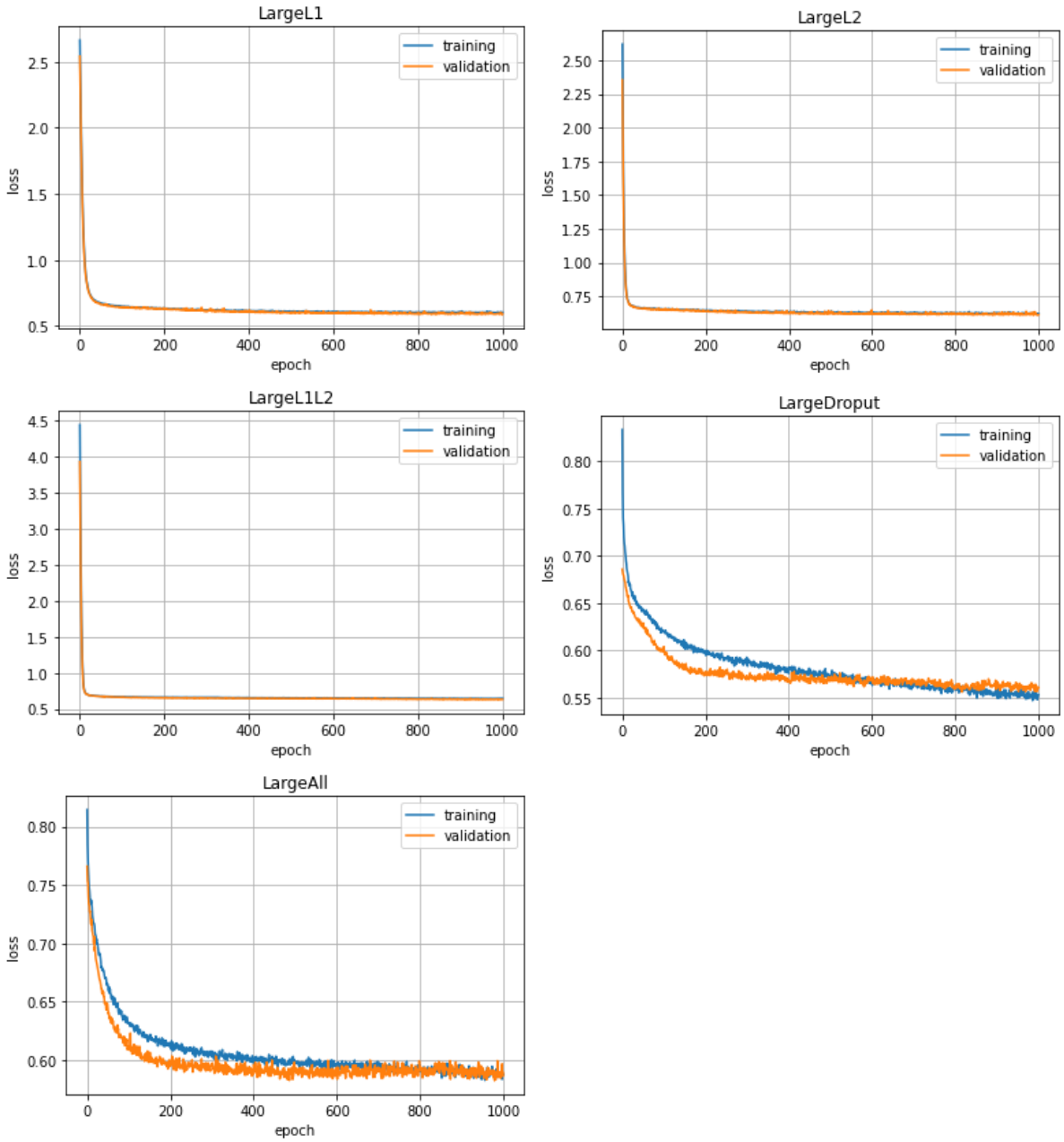
با توجه به اینکه مدل Large بیشترین ظرفیت یادگیری را دارد و اصلا مشکل Underfit برای آن مطرح نیست این مدل را به عنوان مدل برگذیده انتخاب می‌کنیم و سعی می‌کنیم با استفاده از روش‌های منظم‌سازی مشکل Overfit در آن را حل کنیم.

روش‌های منظم سازی بکار رفته:

- منظم‌سازی L1=0.0005
- منطم‌سازی L2=0.005
- ترکیب منظم‌سازی L1=0.0005 و L2=0.005
- افزودن لایه Dropout بعد از هر لایه Dense با احتمال p=0.5
- ترکیب تمام روش‌های بالا p=0.25 و L1=0.00001 و L2=0.0001

نتایج در 1000 بار تکرار در مدل Large:

Model Large	Best Epoch	Train Best	Test Best	Train Last	Test Last
L1=0.0005	947	loss: 0.6091 acc: 0.7011	loss: 0.5971 acc: 0.7140	loss: 0.6041 acc: 0.6757	loss: 0.5914 acc: 0.6950
L2=0.005	992	loss: 0.6382 acc: 0.6825	loss: 0.6234 acc: 0.6940	loss: 0.6214 acc: 0.6460	loss: 0.6133 acc: 0.6450
L1=0.0005 L2=0.005	682	loss: 0.6596 acc: 0.6441	loss: 0.6460 acc: 0.6410	loss: 0.6458 acc: 0.6145	loss: 0.6318 acc: 0.5980
Dropout=0.5	981	loss: 0.5162 acc: 0.7266	loss: 0.5618 acc: 0.7110	loss: 0.5532 acc: 0.6921	loss: 0.5608 acc: 0.6870
L1=0.00001 L2=0.0001 Dropout=0.25	712	loss: 0.5704 acc: 0.7130	loss: 0.5890 acc: 0.7140	loss: 0.5879 acc: 0.6974	loss: 0.5869 acc: 0.6980



- در منظم‌سازی L1 هم‌چنان مقدار بسیار بسیار کمی Overfit داریم و اختلاف دقت در فاز آموزش و ارزیابی حدود ۲ درصد است.
- در منظم‌سازی L2 به هیچ‌عنوان Overfit نداریم اما دقت در فاز آموزش کمی کم‌تر از حد انتظار شده و می‌توان گفت شبکه بسیار بسیار کم دچار Underfit شده‌است. دلیل این اتفاق بزرگ بودن هایپرپارامتر L2 است.
- در ترکیب منظم‌سازی L1 و L2 مشکل Overfit نداریم اما مقادیر دقت بسیار کاهش یافته دلیل این اتفاق این است که ترکیب دو هایپرپارامتر L1 و L2 بیش از حد مدل را منظم کرده (در نمودار دو منحنی val و train یکی شده‌اند.) و اجازه نمی‌دهد مدل یادگیری خوبی داشته باشد در نتیجه دچار مشکل Underfit شدید هستیم.

- در منظم‌سازی Dropout شبکه وضعیت مطلوبی از نظر Overfit و Underfit دارد و مقادیر دقت بسیار خوب هستند. اما مشکلی که در این حالت وجود دارد نواسانات بسیار زیاد در منحنی Loss است که نشان می‌دهد شبکه به خوبی منظم نشده و اثر منظم‌سازی Dropout از L1 و L2 کمتر است.
- در حالت ترکیب L1 و L2 و Dropout برای اینکه شبکه بیش‌از حد منظم نشود و امکان یادگیری از بین برود، هایپرپارامترها را کاهش دادیم. همانطور که از مقادیر داخل جدول نتایج مشخص است بهترین دقت‌ها در این حالت رخ می‌دهد و مقادیر خطا نیز در فاز آموزش و ارزیابی تفاوت چندانی ندارد پس شبکه در بهترین شرایط Fitness است.

✓ بهترین دقت بر روی داده ارزیابی: ۷۱.۴۰ درصد در حالت ترکیب ۳ منظم‌سازی (دقت آموزش ۷۱.۳۰ درصد)

✓ تصویری از محیط Tensor Board: