Understanding SSL without contrastive pairs

Big5 Team

In this article, we summarise the ideas of the paper [1] and try to explain the intuition behind them. The authors present a theoretical study on the dynamics of recent SLL methods which do not use any negative (contrastive) pairs: e.g. BYOL [2] and SimSiam [3]. This study gives insights on how and why these methods work and explains that there is no need for training predictor network via SGD as its weights may be set according to the eigendecomposition of a special correlation matrix.

1 Linear BYOL architecture

The authors study the following simple linear bias-free setting. Suppose we have an input data point $\mathbf{x} \sim p(\mathbf{x})$ and its 2 augmentations $\mathbf{x}_1, \mathbf{x}_2 \sim p(\cdot|\mathbf{x})$. The architecture of the model is presented in Fig. 1. The model consists of 3 networks: *Online*, *Predictor* and *Target*. The corresponding weights matrices are W, W_p and W_a. The *Online* and *Predictor* networks are trained simultaneously, while the *Target* network inherits its weights from the *Online* network with a delay and/or using the Exponential Moving Average (EMA). This is denoted as the *Stop-Gradient* operation at the scheme.

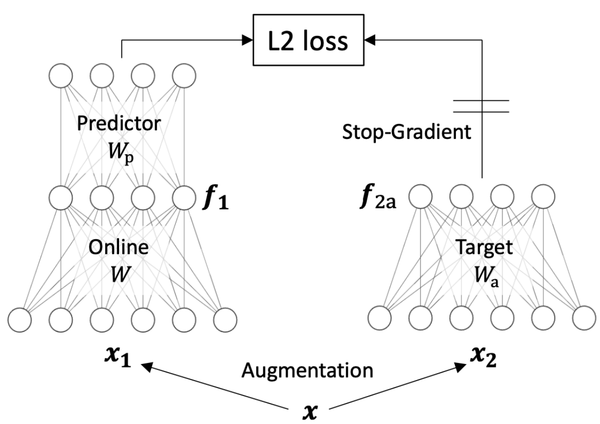

Figure 1: Two-layer setting with a linear, bias-free predictor.

Finally, the objective functional is the following:

$$J(W, W_p) = \frac{1}{2} \cdot \mathbb{E}_{\mathbf{x}_1,\mathbf{x}_2} \left[\| W_p W \mathbf{x}_1 - \mathrm{StopGrad}\left(W_a \mathbf{x}_2\right) \|_2^2 \right] \to \min. \tag{1}$$

2 Gradient update dynamics for BYOL

It is proven that in a limit of large batch sizes and infinitely small learning rates, the gradient update dynamics for the solution of 1 take the following form:

$$\begin{cases} \dot{W}_p = \alpha_p \left(-W W_p \left(X + X'\right) + W_a X\right) W^T - \eta W_p, \\ \dot{W} = W_p^T \left(-W_p W \left(X + X'\right) + W_a X\right) - \eta W, \\ \dot{W}_a = \beta \left(-W_a + W\right); \end{cases} \tag{2}$$

where $X = \mathbb{E}\left[\overline{\mathbf{x}} \cdot \overline{\mathbf{x}}^T\right]$, $\overline{\mathbf{x}}$ is the average augmented view of $\mathbf{x}$, $X' = \mathbb{E}_{\mathbf{x}}\left[\mathbb{V}_{\mathbf{x}'|\mathbf{x}}[\mathbf{x}']\right]$ is the average covariance matrix of the augmented view $\mathbf{x}'$ conditioned on $\mathbf{x}$; α and β are multiplicative learning rate ratios between networks; η is the EMA weight decay parameter. Note that here we assume W, W_a and W_p to be functions of continuous time t.

For SimSiam architecture, we assume $W_a = W$ and do not use EMA.

3 Main theoretical results

The authors formulate and prove a number of theorems that shed some light on the process of BYOL training. The key results are the following.

1. Independent of the target network W_a, the online and predictor networks' weights are balanced by the weight decay parameter η:

$$\forall t > 0 \colon W W^T = \frac{1}{\alpha_p} W_p^T W_p + e^{-2\eta t} C \tag{3}$$

   for some constant symmetric matrix C depended on the initialisation of W and W_p.

2. The stop-gradient is essential for success. In case of no stop-gradient signal, the gradient update dynamics for W will have following form:

$$\frac{d}{dt}\mathrm{vec}(W) = -H(t)\mathrm{vec}(W) \tag{4}$$

   for some some SPD matrix H. Note that if the spectrum of H is bounded from below for any $t > 0$, i.e. $\inf_{t>0} \lambda\left(H(t)\right) \geq \lambda_0 > 0$, then $W(t) \to 0$ as $t \to \infty$. This means that the stop-gradient signal prevents the online network from learning the collapsed (constant zero) representations.

3. Under some assumptions (i.e. $W_a \propto W$; input data points have zero mean and diagonal covariance matrix; symmetric predictor $W_p = W_p^T$), the eigenspaces of W_p and $F = W X W^T$ align. Thus, these matrices are simultaneously diagonalizable:

$$\begin{cases} W_p = U \Lambda_{W_p} U^T, \ \Lambda_{W_p} = \mathrm{diag}(p_1, \ldots, p_d); \\ F = U \Lambda_F U^T, \Lambda_F = \mathrm{diag}(s_1, \ldots, s_d). \end{cases} \tag{5}$$

Consequently, the system 2 decouples into d separate systems of 1D ODEs:

$$\begin{cases} \dot{p}_j = \alpha_p s_j \left(\tau - \left(1+\sigma^2\right) p_j\right) - \eta p_j; \\ \dot{s}_j = 2 p_j s_j \left(\tau - \left(1+\sigma^2\right) p_j\right) - 2\eta s_j, \\ s_j \dot{\tau} = \beta(1-\tau)s_j - \dfrac{\tau \dot{s}_j}{2}; \end{cases} \tag{6}$$

where $\tau(t)$ is the proportion factor ($W_a = \tau W$), σ^2 is the variance of the input ($X' = \sigma^2 I$).

4. The system 6 has an exact integral of motion:

$$s_j(t) = \frac{1}{\alpha_p} p_j^2(t) + e^{-2\eta t} c_j \tag{7}$$

for some constant c_j dependent of the weights initialization.

Thus, the solutions are confined to parabolas of the form $s_j(t) = p_j^2(t) + c_j$ which correspond to the solutions in case of no weight decay ($\eta = 0$). Consequently, the solution for τ has the following form:

$$\tau(t) = \frac{\beta e^{-\beta t}}{p_j(t)} \int_0^t p_j(t') e^{\beta t'} dt'. \tag{8}$$

4 DirectPred: non-trainable predictor network

Based on the theory summarised above, it is proposed to avoid learning weights W_p for the *Predictor* network and set them according to the eigendecomposition of the correlation matrix F. In more detail, the proposed *DirectPred* method includes the following steps:

1. Estimate $F = W X W^T$ by the EMA:

$$\hat{F} = \rho \hat{F} + (1-\rho)\mathbb{E}_B \left[f f^T \right] \tag{9}$$

for the online network output f. Here $\mathbb{E}_B$ denotes expectation w.r.t. batch B.

2. Compute eigendecomposition of $\hat{F}$:

$$\hat{F} = \hat{U} \hat{\Lambda}_F \hat{U}^T, \text{ where } \hat{\Lambda}_F = \operatorname{diag}(s_1, \ldots, s_d). \tag{10}$$

3. Set predictor weights W_p:

$$W_p = \hat{U} \operatorname{diag}(p_1, \ldots, p_d) \hat{U}^T, \text{ where } p_j = \sqrt{s_j} + \varepsilon \max_j s_j. \tag{11}$$

Note that the s_j and p_j (approximately) lie on the desired parabola.

5 Conclusion

References

[1] Y. Tian, X. Chen, and S. Ganguli, "Understanding self-supervised learning dynamics without contrastive pairs," 2021. [Online]. Available: https://arxiv.org/abs/2102.06810

[2] J.-B. Grill, F. Strub, F. Altché, C. Tallec, P. H. Richemond, E. Buchatskaya, C. Doersch, B. A. Pires, Z. D. Guo, M. G. Azar, B. Piot, K. Kavukcuoglu, R. Munos, and M. Valko, "Bootstrap your own latent: A new approach to self-supervised learning," 2020. [Online]. Available: https://arxiv.org/abs/2006.07733

[3] X. Chen and K. He, "Exploring simple siamese representation learning," 2020. [Online]. Available: https://arxiv.org/abs/2011.10566