Understanding self-supervised Learning Dynamics without Contrastive Pairs

Yuandong Tian\textsuperscript{1} \quad Xinlei Chen\textsuperscript{1} \quad Surya Ganguli\textsuperscript{1,2}

\textsuperscript{1} Facebook AI Research \quad \textsuperscript{2} Stanford University

Code: \url{https://github.com/facebookresearch/luckmatters/tree/master/ssl}
Non-contrastive SSL (BYOL/SimSiam)

BYOL: [J. Grill, Bootstrap your own latent: A new approach to self-supervised Learning, NeurIPS 2020]
SimSiam: [X. Chen and K. He, Exploring Simple Siamese Representation Learning, CVPR 2021]
Non-contrastive SSL (BYOL/SimSiam)?

Why do they not collapse to trivial solutions?

\[ x \sim p(\cdot) \]
\[ x_1, x_2 \sim p_{aug}(\cdot|x) \]

Data Augmentation

\[ \text{Dataset} \]

\[ \text{Online } \mathcal{W} \]
\[ \text{Predictor } \mathcal{W}_p \]
\[ \text{Stop-Grad} \]
\[ \text{L2 Loss} \]

No Negative Pairs !!!

BYOL: [J. Grill, Bootstrap your own latent: A new approach to self-supervised Learning, NeurIPS 2020]
SimSiam: [X. Chen and K. He, Exploring Simple Siamese Representation Learning, CVPR 2021]
A simple model

Objective:

\[ J(W, W_p) := \frac{1}{2} \mathbb{E}_{x_1, x_2} [\| W_p f_1 - \text{StopGrad}(f_{2a}) \|^2_2] \]

Linear online network \( W \)

Linear target network \( W_a \)

Linear predictor \( W_p \)
The Dynamics of Training Procedure

**Lemma 1.** BYOL learning dynamics following Eqn. 1:

\[
\begin{align*}
\dot{W}_p &= \alpha_p (-W_p W (X + X') + W_a X) W^\top - \eta W_p \\
\dot{W} &= W_p^\top (-W_p W (X + X') + W_a X) - \eta W \\
\dot{W}_a &= \beta (-W_a + W)
\end{align*}
\]

**Part I** Why we need (1) an **extra predictor** and (2) **stop-gradient**?

**Part II** Why the system doesn’t **collide** to trivial solutions?

**Part III** The role played by different hyperparameters

<table>
<thead>
<tr>
<th>Hyperparameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>(\alpha_p)</td>
<td>Relative learning rate of the predictor</td>
</tr>
<tr>
<td>(\eta)</td>
<td>Weight decay</td>
</tr>
<tr>
<td>(\beta)</td>
<td>The rate of Exponential Moving Average (EMA)</td>
</tr>
</tbody>
</table>

**Part IV** Novel non-contrastive SSL algorithm **DirectPred**

\[\bar{x}(x) := \mathbb{E}_{x' \sim p_{aug}(\cdot|x)} [x']\]

\[X = \mathbb{E} [\bar{x}\bar{x}^\top] \quad \text{Covariance of the data}\]

\[X' = \mathbb{E}_x [\nabla x'|x [x']] \quad \text{Covariance of the augmentation}\]
Part I  No Predictor / No Stop-Gradient do not work

If there is no EMA ($\mathbf{W} = \mathbf{W}_a$), then the dynamics changes:

No Predictor

$$\dot{\mathbf{W}} = -(\mathbf{X}' + \eta \mathbf{I}) \mathbf{W}$$

PSD matrix

No Stop-Gradient (Here $\mathbf{W}_p' := \mathbf{W}_p - \mathbf{I}$)

$$\frac{\text{d}}{\text{d}t} \text{vec}(\mathbf{W}) = - \left[ \mathbf{X}' \otimes (\mathbf{W}_p^T \mathbf{W}_p + \mathbf{I}) + \mathbf{X} \otimes \mathbf{\tilde{W}}_p^T \mathbf{\tilde{W}}_p + \eta \mathbf{I}_{n_1 n_2} \right] \text{vec}(\mathbf{W})$$

PSD matrix

In both cases, $\mathbf{W} \to 0$
Part II Assumptions

**Assumption 1** (Isotropic Data and Augmentation): \( X = I \) and \( X' = \sigma^2 I \)

**Assumption 2**: the EMA weight \( W_a(t) = \tau(t)W(t) \) is a linear function of \( W(t) \)
Symmetrization of the dynamics

**Assumption 3** (Symmetric predictor $W_p$): $W_p(t) = W_p^T(t)$

$W_p$ becomes increasingly symmetric over training

Perfect symmetric $W_p$ might hurt training
Symmetrized Dynamics

Under the three assumptions, the dynamics becomes:

\[
\begin{align*}
\dot{W}_p &= -\frac{\alpha_p}{2}(1 + \sigma^2)\{W_p, F\} + \alpha_p \tau F - \eta W_p \\
\dot{F} &= -(1 + \sigma^2)\{W_p^2, F\} + \tau \{W_p, F\} - 2\eta F
\end{align*}
\]

\(\{A, B\} := AB + BA\) is the anti-commutator.

Here \(F := E[ff^T] = WXW^T\) is the correlation matrix of the input of the predictor \(W_p\). \(F\) is well-defined even with nonlinear network.
Eigenspace Alignment

*Theorem 3*: Under certain conditions,

\[ FW_p - W_p F \to 0 \]

and the eigenspace of \( W_p \) and \( F \) gradually aligns.
Why non-contrastive SSL doesn’t collapse?

When eigenspace aligns, the dynamics becomes decoupled:

\[
\begin{align*}
\dot{p}_j &= \alpha_p s_j \left[ \tau - (1 + \sigma^2) p_j \right] - \eta p_j \\
\dot{s}_j &= 2 p_j s_j \left[ \tau - (1 + \sigma^2) p_j \right] - 2 \eta s_j \\
\dot{s}_j \dot{\tau} &= \beta (1 - \tau) s_j - \tau \dot{s}_j / 2.
\end{align*}
\]

Where \(p_j\) and \(s_j\) are eigenvalues of \(W_p\) and \(F\)

Invariance holds: \(s_j(t) = \alpha_p^{-1} p_j^2(t) + e^{-2\eta t} c_j\)
Why non-contrastive SSL doesn’t collapse?

1D dynamics of the eigenvalue $p_j$ of $W_p$:

$$\dot{p}_j = p_j^2 \left[ \tau(t) - (1 + \sigma^2)p_j \right] - \eta p_j$$

- EMA
- Variance due to data augmentation
- Weight Decay
Why non-contrastive SSL doesn’t collapse?

1D dynamics of the eigenvalue $p_j$ of $W_p$:

$$
\dot{p}_j = p_j^2 \left[ \tau(t) - (1 + \sigma^2)p_j \right] - \eta p_j
$$

- **EMA**
- **Variance due to data augmentation**
- **Weight Decay**

**Stable stationary point**  
**Unstable stationary point**

---

**EMA**

Variance due to data augmentation  
Weight Decay

---

**Stable Trivial**  
**Stable Nontrivial**

---

**Stable stationary point**  
**Unstable stationary point**
Why non-contrastive SSL doesn’t collapse?

1D dynamics of the eigenvalue $p_j$ of $W_p$:

$$\dot{p}_j = p_j^2 \left[ \tau(t) - (1 + \sigma^2)p_j \right] - \eta p_j$$

- EMA
- Variance due to data augmentation
- Weight Decay

Stable Trivial Basin

Non-trivial Basin

Stable stationary point

Unstable stationary point

$$p_{j-} = \frac{\tau - \sqrt{\tau^2 - 4\eta(1 + \sigma^2)}}{2(1 + \sigma^2)} \sim \frac{\eta}{\tau}$$
Part III The Effect of Weight Decay $\eta$

(a) No Weight Decay

(b) Weak Weight Decay

(c) Strong Weight Decay

$\eta = 0$

$\eta \sim \frac{\eta}{\tau}$

$\eta < \frac{\tau^2}{4(1 + \sigma^2)}$

$\eta > \frac{\tau^2}{4(1 + \sigma^2)}$

• Stable stationary point
• Unstable stationary point
The Benefit of Weight Decay

Eigenspace alignment condition

\[ p_j[\tau - (1 + \sigma^2)p_j] < \frac{1}{2} [\alpha p (1 + \sigma^2)s_j + 3\eta] \]

Higher weight decay \( \rightarrow \) alignment condition is more likely to satisfy!
Relative learning rate of the predictor $\alpha_p$

**Positive 🌟**
1. Large $\alpha_p$ shrinks the size of trivial basin
2. Relax the condition of eigenspace alignment

**Negative 😞** With very large $\alpha_p$, eigenvalue of $F$ won’t grow (and no feature learning)
Exponential Moving Average rate $\beta$

$\beta$ large $\rightarrow W_a(t)$ catches $W(t)$ faster $\rightarrow \tau$ grows faster to 1

**Positive 😊:** Slower rate (small $\beta$) relaxes the condition of eigenspace alignment

**Negative 😞:** Slower rate makes the training slow and expands the size of trivial basin
Part IV DirectPred

• Directly setting linear $W_p$ rather than relying on gradient update.

1. Estimate $\hat{F} = \rho \hat{F} + (1 - \rho)E[\mathbf{f}\mathbf{f}^T]$
2. Eigen-decompose $\hat{F} = \hat{U}\Lambda_F\hat{U}^T$, $\Lambda_F = \text{diag} [s_1, s_2, \ldots, s_d]$
3. Set $W_p$ following the invariance:

\[ p_j = \sqrt{s_j} + \epsilon \max_j s_j, \quad W_p = \hat{U}\text{diag}[p_j]\hat{U}^T \]

Guaranteed Eigenspace Alignment 😊
## Performance of DirectPred on STL-10/CIFAR-10

<table>
<thead>
<tr>
<th>Downstream Classification Top-1</th>
<th>Number of epochs</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>100</td>
</tr>
<tr>
<td></td>
<td>STL-10</td>
</tr>
<tr>
<td>DirectPred</td>
<td>77.86 ± 0.16</td>
</tr>
<tr>
<td>DirectPred (freq=5)</td>
<td>77.54 ± 0.11</td>
</tr>
<tr>
<td>SGD baseline</td>
<td>75.06 ± 0.52</td>
</tr>
<tr>
<td></td>
<td>CIFAR-10</td>
</tr>
<tr>
<td>DirectPred</td>
<td><strong>85.21 ± 0.23</strong></td>
</tr>
<tr>
<td>DirectPred (freq=5)</td>
<td>84.93 ± 0.29</td>
</tr>
<tr>
<td>SGD baseline</td>
<td>84.49 ± 0.20</td>
</tr>
</tbody>
</table>
### Performance of DirectPred on ImageNet

#### Downstream classification (ImageNet):

<table>
<thead>
<tr>
<th>BYOL variants</th>
<th>Accuracy (60 ep)</th>
<th></th>
<th>Accuracy (300 ep)</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Top-1</td>
<td>Top-5</td>
<td>Top-1</td>
<td>Top-5</td>
</tr>
<tr>
<td>2-layer predictor</td>
<td>64.7</td>
<td>85.8</td>
<td>72.5</td>
<td>90.8</td>
</tr>
<tr>
<td>linear predictor</td>
<td>59.4</td>
<td>82.3</td>
<td>69.9</td>
<td>89.6</td>
</tr>
<tr>
<td>DirectPred</td>
<td>64.4</td>
<td>85.8</td>
<td>72.4</td>
<td>91.0</td>
</tr>
</tbody>
</table>

* 2-layer predictor is BYOL default setting.

DirectPred using linear predictor is better than SGD with linear predictor, and is comparable with 2-layer predictor.
Conclusion

• A systematic analysis on the dynamics of non-contrastive self-supervised learning (SSL) methods
  • **Part I** Why we need (1) an **extra predictor** and (2) **stop-gradient**?
  • **Part II** Why training doesn’t **collapse** to trivial solutions?
  • **Part III** The role played by different hyperparameters

• Propose **DirectPred**, a novel non-contrastive SSL method
  • Directly align the eigenspace of the predictor $W_p$ with the correlation matrix $F$
  • Comparable performance in downstream classification tasks, compared to vanilla BYOL
    • CIFAR-10/STL-10
    • ImageNet (60 epochs / 300 epochs)

Code: [https://github.com/facebookresearch/luckmatters/tree/master/ssl](https://github.com/facebookresearch/luckmatters/tree/master/ssl)
Thanks!