Understanding Contrastive Representation Learning through Alignment and Uniformity on the Hypersphere

Tongzhou Wang & Phillip Isola
MIT CSAIL

Presented by Shijing Si, Figures and Tables are adapted from their paper
Background: Representation on the Unit Hypersphere

Many empirical works map data to unit hypersphere representation space. [Parkhi et al. 2015; Bojanowski & Joulin 2017; Wu et al. 2018; Bachman et al. 2019; Tian et al. 2019; He et al. 2019, Chen et al. 2020; ...]

Intuitively, fixed-norm feature vectors can

- improve training stability [Xu & Durrett 2018; Wang et al. 2017].
- make well-clustered sets linearly separable.

But, are all $l_2$-normalized encoders created equal?
In this work, authors...

- Propose optimizable metrics for two desirable representation properties: *Alignment* and *Uniformity*, with theoretical motivations connecting with the contrastive loss.

- Experimentally demonstrate the causal relation between these properties and representation quality (*i.e.*, downstream task performance), and show that directly optimizing them leads to comparable or better quality.
What is a good representation?

**Invariant to random noise factors**

E.g., image features should be invariant to photon shot noise

Often implemented as requiring features for a positive pair (2 random aug. of a sample) to be similar

**Preserve as much information as possible**

I.e., InfoMax of feature distribution

Not knowing the downstream task beforehand, features should keep information in data
What is a good representation?

**Alignment:** Similar samples have similar features

**Positive Pair:** $(x, y)$ from $p_{pos}$

- $f(x)$
- $f(y)$

Preserve as much information as possible

I.e., InfoMax of feature distribution

Not knowing the downstream task beforehand, features should keep information in data
What is a good representation?

**Alignment:** Similar samples have similar features

**Uniformity:** Preserve maximal information
What is a good representation?

Contrastive loss pulls positive pairs together, pushes negatives apart.

Intuitively, it seems optimizing *Alignment* and *Uniformity*.

Does it in practice?
Feature distribution from Contrastive Learning

**Toy example:**

Train CIFAR-10 encoders with $S^1$ feature space (circle). Visualize feature distributions on the validation set.

---

**Alignment**

Positive Pair Feature Distances

**Uniformity**

Feature Distribution

- Gaussian KDE on feature vectors
- vMF KDE on angles

Unsupervised Contrastive Learning
Feature distribution from Contrastive Learning

**Toy example:**
Train CIFAR-10 encoders with $S^1$ feature space (circle).
Visualize feature distributions on the validation set.

**Unsupervised Contrastive Learning**
**Supervised Predictive (NLL) Learning**
**Random Network Initialization**
Feature distribution from Contrastive Learning

Toy example:
Train CIFAR-10 encoders with $S^1$ feature space (circle).
Visualize feature distributions on the validation set.

Unsupervised Contrastive Learning
Supervised Predictive (NLL) Learning
Random Network Initialization

Aligned and Uniform feature distribution!
Feature distribution from Contrastive Learning

Now,

Can we quantify Alignment and Uniformity?

Can we formally show their relation with the contrastive loss?
Background: Contrastive Learning

Data:

\[ p_{\text{data}}(\cdot) : \text{data distribution over } \mathbb{R}^n \]
\[ p_{\text{pos}}(\cdot) : \text{positive pair distribution over } \mathbb{R}^n \times \mathbb{R}^n \]

\[ \forall x, y \quad p_{\text{pos}}(x, y) = p_{\text{pos}}(y, x) \quad \text{(symmetry)} \]

\[ \forall x \quad \int p_{\text{pos}}(x, y) = p_{\text{data}}(x) \quad \text{(matching marginal)} \]

Encoder:

\[ f : \mathbb{R}^n \rightarrow S^{d-1} \]

Objective:

\[ \mathcal{L}_{\text{contrastive}}(f; \tau, M) \triangleq \mathbb{E}_{(x, y) \sim p_{\text{pos}}, \{x_i\}_{i=1}^M \sim p_{\text{data}}} \left[ -\log \frac{e^{f(x)^T f(y)/\tau}}{e^{f(x)^T f(y)/\tau} + \sum_i e^{f(x_i)^T f(y)/\tau}} \right] \]
Metrics for Alignment and Uniformity

**Alignment:** expected positive pair feature distance

\[ \mathcal{L}_{\text{align}}(f; \alpha) \triangleq \mathbb{E}_{(x, y) \sim p_{\text{pos}}} \left[ \| f(x) - f(y) \|_2^\alpha \right] \quad \alpha > 0 \]

**Uniformity:** logarithm of expected pairwise Gaussian potential

\[ \mathcal{L}_{\text{uniform}}(f; t) \triangleq \log \mathbb{E}_{x, y \sim \text{i.i.d.} \ \text{pdata}} \left[ G_t(f(x), f(y)) \right] \triangleq \log \mathbb{E}_{x, y \sim \text{i.i.d.} \ \text{pdata}} \left[ e^{-t \| f(x) - f(y) \|_2^2} \right] \quad t > 0 \]
Uniformity: Gaussian (RBF) Kernel

\[ G_t : S^{d-1} \times S^{d-1} \rightarrow \mathbb{R}, \] defined as \( G_t(u, v) \triangleq e^{-t\|u-v\|^2} = e^{2t u^T v - 2t}, \quad t > 0 \)

For distribution \( p \) over a unit hypersphere, the average pairwise Gaussian potential \( E_{u, v} \sim \mathcal{U} \left[ G_t(u, v) \right] \) is a good metric for uniformity.

Empirically,

Theoretically,
- Points minimizing \( E_{u, v} \left[ G_t(u, v) \right] \) asymptotically converge to the uniform distribution.
- Distribution minimizing \( E_{u, v} \left[ G_t(u, v) \right] \) is uniquely the uniform distribution.
Asymptotics of $\mathcal{L}_{\text{contrastive}}$

**Theorem 1 (Asymptotics of $\mathcal{L}_{\text{contrastive}}$).** For fixed $\tau > 0$, as the number of negative samples $M \to \infty$, the (normalized) contrastive loss converges to

$$
\lim_{M \to \infty} \mathcal{L}_{\text{contrastive}}(f; \tau, M) - \log M
= \lim_{M \to \infty} \mathbb{E}_{(x, y) \sim p_{\text{pos}}, \{x_i^+\}_{i=1}^{\infty} \sim p_{\text{data}}} \left[ -\log \frac{e^{f(x)^T f(y)/\tau}}{e^{f(x)^T f(y)/\tau} + \sum_i e^{f(x_i^+)^T f(y)/\tau}} \right] - \log M
= -\frac{1}{\tau} \mathbb{E}_{(x, y) \sim p_{\text{pos}}} [f(x)^T f(y)] + \mathbb{E}_{x \sim p_{\text{data}}} \left[ \log \mathbb{E}_{x^- \sim p_{\text{data}}} [e^{f(x^-)^T f(x)/\tau}] \right].
$$

(2)

We have the following results:

1. The first term is minimized iff $f$ is perfectly aligned.

Perfect alignment: Optimal $\mathcal{L}_{\text{align}}$, mapping positive pairs to same features.
Asymptotics of $\mathcal{L}_{\text{contrastive}}$

**Theorem 1** (Asymptotics of $\mathcal{L}_{\text{contrastive}}$). For fixed $\tau > 0$, as the number of negative samples $M \to \infty$, the (normalized) contrastive loss converges to

$$
\lim_{M \to \infty} \mathcal{L}_\text{contrastive}(f; \tau, M) - \log M = \lim_{M \to \infty} \mathbb{E}_{(x,y) \sim p_{\text{pos}} \cup \{x_i\}_{i=1}^M \sim p_{\text{data}}} \left[ - \log \frac{e^{f(x)^T f(y)/\tau}}{e^{f(x)^T f(y)/\tau} + \sum_i e^{f(x_i)^T f(y)/\tau}} \right] - \log M
$$

$$=-\frac{1}{\tau} \mathbb{E}_{(x,y) \sim p_{\text{pos}}} [f(x)^T f(y)] + \mathbb{E}_{x \sim p_{\text{data}}} \left[ \log \mathbb{E}_{x^+ \sim p_{\text{data}}} \left[ e^{f(x^-)^T f(x)/\tau} \right] \right].$$

(2)

We have the following results:

1. The first term is minimized iff $f$ is perfectly aligned.
2. If perfectly uniform encoders exist, they form the exact minimizers of the second term.

**Perfect alignment:** Optimal $\mathcal{L}_{\text{align}}$, mapping positive pairs to same features.

**Perfect uniformity:** Uniform feature distribution.
Asymptotics of $\mathcal{L}_{\text{contrastive}}$

$$\lim_{M \to \infty} \mathcal{L}_{\text{contrastive}}(f; \tau, M) - \log M = -\frac{1}{\tau} \mathbb{E}_{(x, y) \sim \rho_{\text{con}}} \left[ f(x)^\top f(y) \right] + \mathbb{E}_{x \sim \rho_{\text{data}}} \log \mathbb{E}_{x^- \sim \rho_{\text{data}}} \left[ e^{f(x^-)^\top f(x)/\tau} \right]$$  

(Thm. 1)
Asymptotics of $\mathcal{L}_{\text{contrastive}}$

$$\lim_{M \to \infty} \mathcal{L}_{\text{contrastive}}(f; \tau, M) - \log M = -\frac{1}{T} \mathbb{E}_{(x,y) \sim \text{Post}} \left[ f(x)^T f(y) \right] + \mathbb{E}_{x \sim \text{Data}} \left[ \log \mathbb{E}_{z \sim \text{Data}} \left[ e^{f(x)^T f(z)/\tau} \right] \right]$$

(Thm. 1)

- **InfoMax**: $\max_f I(f(x); f(y)) = \max_f -H(f(x)|f(y)) + H(f(x))$ (max MI between two views).
Asymptotics of $\mathcal{L}_{\text{contrastive}}$

$$\lim_{M \to \infty} \mathcal{L}_{\text{contrastive}}(f; \tau, M) - \log M = \frac{1}{T(x,y) \sim \text{Post}} \mathbb{E} \left[ f(x)^\top f(y) \right] + \mathbb{E}_{x \sim \text{Data}} \log \mathbb{E}_{x' \sim \text{Data}} \left[ e^{f(x')^\top f(x)/\tau} \right]$$  

(Thm. 1)  

- $\mathcal{L}_{\text{contrastive}}$ optimizes for *aligned* and *information-preserving* encoders.
Experiments
Relation Between Representation Quality and Alignment & Uniformity

We train multiple encoders on
- STL-10
- NYU-Depth-V2
- ImageNet-100 subset (MoCo-based)
- BookCorpus (Quick-Thought-Vectors-based)
Relation Between Representation Quality and Alignment & Uniformity

We train multiple encoders on
- STL-10
- NYU-Depth-V2
- ImageNet-100 subset (MoCo-based)
- BookCorpus (Quick-Thought-Vectors-based)

with varying
- **Objective**: (possibly zero) weighted combination of $L_{\text{contrastive}}$, $L_{\text{uniform}}$, and $L_{\text{align}}$
- **Batch size**: affecting #pairs in $L_{\text{contrastive}}$ and $L_{\text{uniform}}$
- **Feature dimensionality**
- **Loss hyperparameters**: $\tau, \alpha, t$
- **Training schedule** (#epochs, LR)
- **Initialization** (from scratch vs. pretrained).
Relation Between Representation Quality and Alignment & Uniformity
Relation Between Representation Quality and Alignment & Uniformity
Relation Between Representation Quality and Alignment & Uniformity

Among encoders with 128 dim and 768 batch size

- Best encoder trained with $\mathcal{L}_{\text{align}}$ and $\mathcal{L}_{\text{uniform}}$: 81.15%
- Best encoder trained with $\mathcal{L}_{\text{contrastive}}$: 80.46%

* Best encoder is selected by cross-validation on training set.
Relation Between Representation Quality and Alignment & Uniformity

![Graphs showing linear classification outputs and 5-NN classification on fc7, highlighting effects of different loss functions on accuracy and uniformity.](image-url)
## Comparable Performance on STL-10

| Loss Formula | Validation Set Accuracy ↑ | Output + Linear | Output + 5-NN | fc7 + Linear | fc7 + 5-NN |
|--------------|---------------------------|----------------|--------------|--------------|------------|
| $\mathcal{L}_{\text{contrastive}}(\tau = 0.19)$ | 80.46% | 78.75% | 83.89% | 76.33% |
| Best $\mathcal{L}_{\text{align}}$ and $\mathcal{L}_{\text{uniform}}$ only | 81.15% | 78.89% | 84.43% | 76.78% |
| Best among all encoders | 81.06% | 79.05% | 84.14% | 76.48% |

Table 1: STL-10 encoder evaluations. Numbers show linear and 5-nearest neighbor (5-NN) classification accuracies on the validation set. The best result is picked by encoder outputs linear classifier accuracy from a 5-fold training set cross validation, among all 150 encoders trained from scratch with 128-dimensional output and 768 batch size.
Relation Between Representation Quality and Alignment & Uniformity

64 NYU-Depth-V2 Encoders

Depth Prediction on conv5

Depth Prediction on conv4

GOOD!!!

GOOD!!!
## Comparable Performance on NYU-Depth

| Loss Formula                                      | Validation Set MSE ↓ |   |
|---------------------------------------------------|----------------------|--|
| Best $\mathcal{L}_{contrastive}$ only             | 0.5 · $\mathcal{L}_{contrastive}(\tau=0.1)$ | 0.7024 | **0.7575** |
| Best $\mathcal{L}_{align}$ and $\mathcal{L}_{uniform}$ only | 0.75 · $\mathcal{L}_{align}(\alpha=2)$ + 0.5 · $\mathcal{L}_{uniform}(t=2)$ | **0.7014** | 0.7592 |
| Best among all encoders                           | 0.75 · $\mathcal{L}_{align}(\alpha=2)$ + 0.5 · $\mathcal{L}_{uniform}(t=2)$ | **0.7014** | 0.7592 |

Table 2: NYU-DEPTH-V2 encoder evaluations. Numbers show depth prediction mean squared error (MSE) on the validation set. The best result is picked based on conv5 layer MSE from a 5-fold training set cross validation, among all 64 encoders trained from scratch with 128-dimensional output and 128 batch size.
Optimize the weights of assignment and uniformity

as long as the ratio between two weights is not too large (e.g., < 4), we observe that the representation quality remains relatively good and insensitive to the exact weight choices.

Figure 7: Effect of optimizing different weighted combinations of $L_{\text{align}}(\alpha=2)$ and $L_{\text{uniform}}(t=2)$ for STL-10. For each encoder, we show the $L_{\text{align}}$ and $L_{\text{uniform}}$ metrics, and validation accuracy of a linear classifier trained on encoder outputs. $L_{\text{uniform}}$ is exponentiated for plotting purposes.
Correlation or Causation?

Take an STL-10 encoder trained w.r.t. only $L_{\text{contrastive}}$ and a suboptimal temperature $\tau=2.5$.

Fine-tune according to $L_{\text{align}}$ and/or $L_{\text{uniform}}$.

Representation quality degrades if only one property is optimized.
Correlation or Causation?

Take an STL-10 encoder trained w.r.t. only $\mathcal{L}_{\text{contrastive}}$ and a suboptimal temperature $T=2.5$. Fine-tune according to $\mathcal{L}_{\text{align}}$ and/or $\mathcal{L}_{\text{uniform}}$.

Representation quality degrades if only one property is optimized. Representation quality improves if both properties are optimized.
$\mathcal{L}_{\text{align}} \& \mathcal{L}_{\text{uniform}}$ Implementation

• Simple implementation

```python
# bsz : batch size (number of positive pairs)
# d  : latent dim
# x  : Tensor, shape=[bsz, d]
# y  : Tensor, shape=[bsz, d]
# latents for one side of positive pairs
# latents for the other side of positive pairs

def align_loss(x, y, alpha=2):
    return (x - y).norm(p=2, dim=1).pow(alpha).mean()

def uniform_loss(x, t=2):
    return torch.pdist(x, p=2).pow(2).mul(-t).exp().mean().log()
```

PyTorch code

• Comparable or better performance than $\mathcal{L}_{\text{contrastive}}$

• Code: [github.com/SsnL/align_uniform](https://github.com/SsnL/align_uniform)

  ✓ Losses
  ✓ STL-10 example
  ✓ MoCo with $\mathcal{L}_{\text{align}}$ and $\mathcal{L}_{\text{uniform}}$
Discussion & Implications

• Why is the unit hypersphere a nice representation space?

   A lot of intuitions, but no rigorous theory so far.

• Beyond contrastive learning.

   Many other representation learning methods uses the unit hypersphere (e.g., NoiseAsTarget [Bojanowski & Joulin 2017]). Is it possible to relate them to alignment and/or uniformity?

• Other desirable representation properties.

   Are there other desirable representation properties? How can we enforce them?