Toward Understanding the Feature Learning Process of Self-supervised Contrastive Learning

Zixin Wen [1] **Yuanzhi Li** [2]

Abstract

We formally study how contrastive learning learns the feature representations for neural networks by analyzing its feature learning process. We consider the case where our data are comprised of two types of features: the more semantically aligned sparse features which we want to learn from, and the other dense features we want to avoid. Theoretically, we prove that contrastive learning using **ReLU** networks provably learns the desired sparse features if proper augmentations are adopted. We present an underlying principle called **feature decoupling** to explain the effects of augmentations, where we theoretically characterize how augmentations can reduce the correlations of dense features between positive samples while keeping the correlations of sparse features intact, thereby forcing the neural networks to learn from the self-supervision of sparse features. Empirically, we verified that the feature decoupling principle matches the underlying mechanism of contrastive learning in practice.

1. Introduction

Self-supervised learning (Devlin et al., 2019; Mikolov et al., 2013; Sutskever et al., 2014; Jing & Tian, 2020) has demonstrated its immense power in different areas of machine learning (e.g. BERT (Devlin et al., 2019) in natural language processing). Recently, it has been discovered that contrastive learning (e.g., Tian et al. (2019); He et al. (2020); Chen et al. (2020a); Chen et al. (2020); Grill et al. (2020); Chen & He (2020)), one of the most typical forms of self-supervised learning, can indeed learn representations of image data that achieve superior performance in many downstream vision tasks. Moreover, as shown by the seminal work (He et al., 2020), the learned feature representations can even outperform those learned by supervised learning in several downstream tasks. The remakable potential of contrastive learning methods poses challenges for researchers to understand and improve upon such simple but effective algorithms.

Contrastive learning in vision learns the feature representations by minimizing pretext task objectives similar to the cross-entropy loss used in supervised learning, where both the inputs and "labels" are derived from the unlabeled data, especially by using augmentations to create multiple views of the same image. The seminal paper Chen et al. (2020b) has demonstrated the effects of stronger augmentations (comparing to supervised learning) for the improvement of feature quality. Tian et al. (2020a) showed that as the augmentations become stronger, the quality of representations displayed a U-shaped curve. Such observations provided insights into the inner-workings of contrastive learning. But it remains unclear *what has happened in the learning process* that renders augmentations necessary for successful contrastive learning.

Some recent works have been done to understand contrastive learning from theoretical perspective (Arora et al., 2019; Wang & Isola, 2020; Tsai et al., 2020). However, these works have not analyzed how **data augmentations** affect the **feature learning process** in **neural networks**, which we deem as crucial to understand how contrastive learning works in practice. We state the fundamental questions we want to address below, and provide tentative answers to all the questions by building theory on a simplified model that shares similar structures with real scenarios, and we provide some empirical evidence through experiments to verify the validity of our models.

Fundamental Questions

1. How do **neural networks** trained by contrastive learning learn their feature representations **efficiently**, and are the representations similar to those learned in supervised learning?
2. Why does contrastive learning in deep learning collapse in practice when no augmentation is used, and how do standard data augmentations help contrastive learning?

[1]University of International Business and Economics, Beijing [2]Carnegie Mellon University. Correspondence to: Zixin Wen <zixinw@andrew.cmu.edu>, Yuanzhi Li <yuanzhil@andrew.cmu.edu>.

Proceedings of the 38^{th} International Conference on Machine Learning, PMLR 139, 2021.

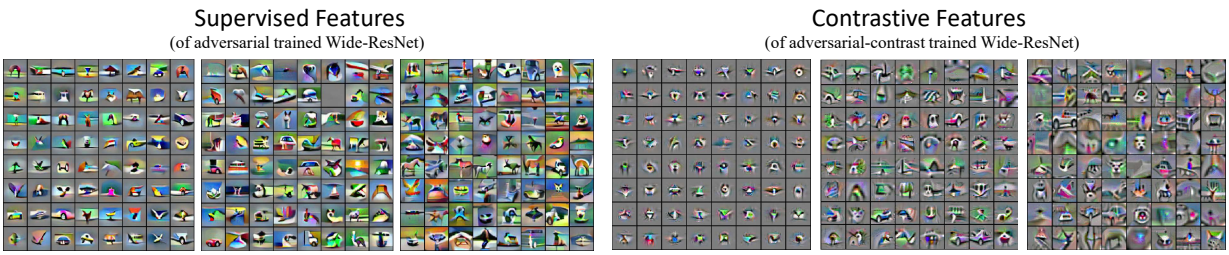


Figure 1. The difference between supervised features and contrastive features (in the higher layers of Wide-ResNet 34x5 over CIFAR10). While both features contain shapes of objects, the supervised features are more colorful than the contrastive features. (here both crop-resize and color distortion were used in contrastive learning, while no color distortion was used in supervised learning. The adversarial-contrast learning follows Kim et al. (2020)). And we use the visualization technique in Allen-Zhu & Li (2020b).

1.1. Our Contributions

In this paper we directly analyze the **feature learning process** of contrastive learning for neural networks (i.e. learning the hidden layers of the neural network). Our results hold for certain data distributions based on *sparse coding model*. Mathematically, we assume our input data are of the form $x = \mathbf{M}z + \xi$, where $\mathbf{M}z$ is called the sparse signal such that $\|z\|_0 = \widetilde{O}(1)$, and ξ is **the spurious dense noise**, where we simply assume that ξ follows from certain dense distributions (such that $\mathrm{span}(\xi) \equiv \mathrm{span}(x)$) with large norm (e.g., $\|\xi\|_2 = \mathsf{poly}(d) \gg \|\mathbf{M}z\|_2 \approx \widetilde{O}(1)$). Formal definition will be presented in Section 2, as we argue that sparse coding model is indeed a proper *provisional* model to study the feature learning process of contrastive learning over the given data set.

Theoretical results. Over our data distributions based on sparse coding model, when we perform contrastive learning by using stochastic gradient descent (SGD) to train a one-hidden-layer neural networks with ReLU activations:

- If no augmentation is applied to the data inputs, *the neural networks will learn feature representations that emphasize the spurious dense noise*, which can easily overwhelm the sparse signals.
- If *natural* augmentation techniques (in particular, the random mask defined in Definition 2.3) are applied to the training data, *the neural networks will avoid learning the features associated with dense noise but pick up the features on the sparse signals*. Such a difference of features brought by augmentation is due to a principle we refer to as **"feature decoupling"**. Moreover, these features can be learned *efficiently* simply by doing a variant of Stochastic Gradient Descent (SGD) over the contrastive training objective (after data augmentations).
- The features learned by neural networks via contrastive learning (with augmentations) is similar to the features learned via supervised learning (under sparse coding model). This claim holds as long as two requirements are satisfied: (1) The sparse signals in the data have not been corrupted by augmentations in contrastive learning; (2) The labels in supervised learning depends mostly on the sparse signals.

Therefore, our theory indicates that in our model, the success of contrastive learning of neural networks relies essentially on the data augmentations to remove the features associated with the spurious dense noise. We abstract this process into a principle below, which we show to hold in neural networks used in real-world settings as well.

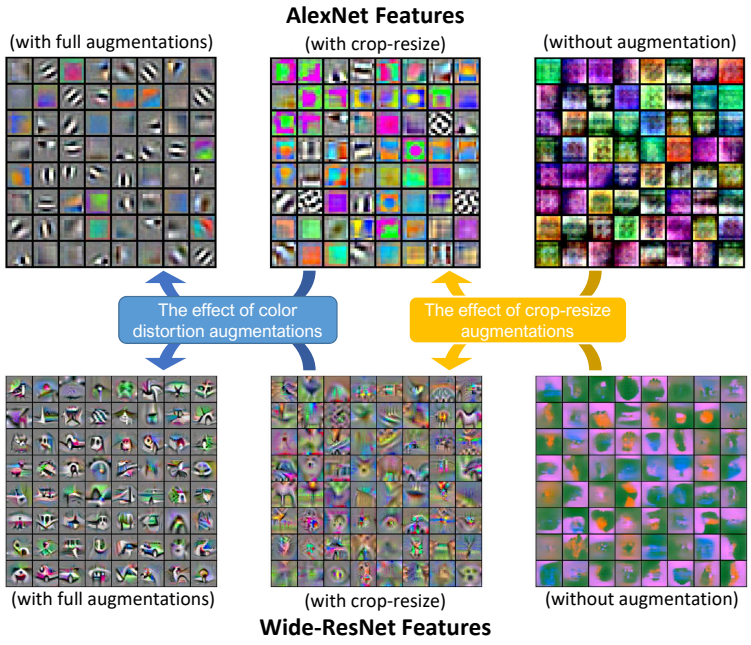
Feature Decoupling

Augmentations in contrastive learning serve to **decouple the correlations of spurious features** between the representations of positive samples. Moreover, after the augmentations, the neural networks will ignore the decoupled features and learn from the similarities of features that are *more resistant to data augmentations*.

We will prove that contrastive learning can successfully learn the desired sparse features using this principle. The intuitions of our proof will be present in Section 4.

Empirical evidence of our theory. Empirically, we conduct multiple experiments to justify our theoretical statements, and the results indeed matches our theory. We show:

- **When no proper augmentation is applied to the data, the neural network will learn features with dense patterns.** As shown in Figure 2, Figure 3 and Figure 4: If no augmentations are used, the learned features are completely meaningless and the representations are dense; If only crop-resize augmentations are used, then the mixture of color features (which also generate dense firing patterns) will remain in the neural network and prevent further separation of clusters.



Message ❶: If no augmentations are applied to the inputs, *the weights of neural nets will not stay at random initialization.* **Contrastive learning is still performing feature learning without augmentation, but these features emphasize on the spurious noise instead of the true signals.**

Message ❷: Applying only crop-resize augmentations can help neural networks learn some of the semantically meaningful features, but it fail to remove the dense mixture of color features learned by contrastive learning.

Message ❸: Applying color distortion augmentations can help neural networks to remove some of the dense mixture of color features so that the edge (and shape) features are learned with much better quality.

Figure 2. Evidence of **feature decoupling**: how do augmentations affect the features learned by neural networks in contrastive learning. The two different augmentations we have conducted here are *color distortions* and *crop-resize*.The color distortions we used consist of color jittering and random grayscale.

- **Standard augmentations removes features associated with dense patterns, and the remaining features learned by contrastive learning do exhibit sparse firing pattern.** As shown in Figure 3 and Figure 4, if no (suitable) augmentations are applied, the neural networks will learn dense representations of image data. After the augmentations, neural networks will successfully form separable clusters of representations for image data, and the *learned features indeed emphasizes sparse signals*.
- **The features learned in contrastive learning resemble the features learned in supervised learning**. As shown in Figure 1, the shape features (filters that exhibit shape images) of the higher layer of Wide ResNet via supervised learning are similar to those learned in contrastive learning. However, color features learned in supervised learning are much more than those in contrastive learning. This verifies our theoretical results that features preserved under augmentations will be learned by both contrastive and supervised learning.

1.2. Related Work

Self-supervised learning. Self/un-supervised representation learning has a long history in the literature. In natural language processing (NLP), self-supervised learning has been the major approach (Mikolov et al., 2013; Devlin et al., 2019). In vision, the generative approach has been the favored approach (Radford et al., 2016; Arjovsky et al., 2017; Kingma & Welling, 2014). The initial works (Carreira-Perpiñán & Hinton, 2005; Smith & Eisner, 2005; Gutmann & Hyvärinen, 2012) of contrastive learning focus on learning the hidden latent variables of the data. Later the attempts to use self-supervised to help pretraining brought the contrastive learning to vision (Oord et al., 2018; Tian et al., 2019; He et al., 2020; Chen et al., 2020a; Chen et al., 2020; Grill et al., 2020; Chen & He, 2020). On the theoretical side, there has been a lot of papers trying to understand un/self-supervised learning (Coates et al., 2011; Radhakrishnan et al., 2018; Arora et al., 2019; Nguyen et al., 2019; Lee et al., 2020; Wang & Isola, 2020; Tsai et al., 2020; Tian et al., 2020a; Tosh et al., 2020; 2021). For contrastive learning, Arora et al. (2019) assume that different positive samples are independently drawn from the same latent class, which reduce the problem to supervised learning. Wang & Isola (2020) pointed out the tradeoff between alignment and uniformity. Tsai et al. (2020); Tian et al. (2020a) proposed to analyze contrastive learning from information-theoretic framwork. Lee et al. (2020) used a pretext task similar to generative approach, but is restricted to the linear models. These papers did not discuss *how features are learned* in **neural networks** and *how augmentations affect the learned features*, which are essential to understand contrastive learning in practice. (Tian et al., 2020b) tried to analyze the learning process, but their augmentation can fix the class-

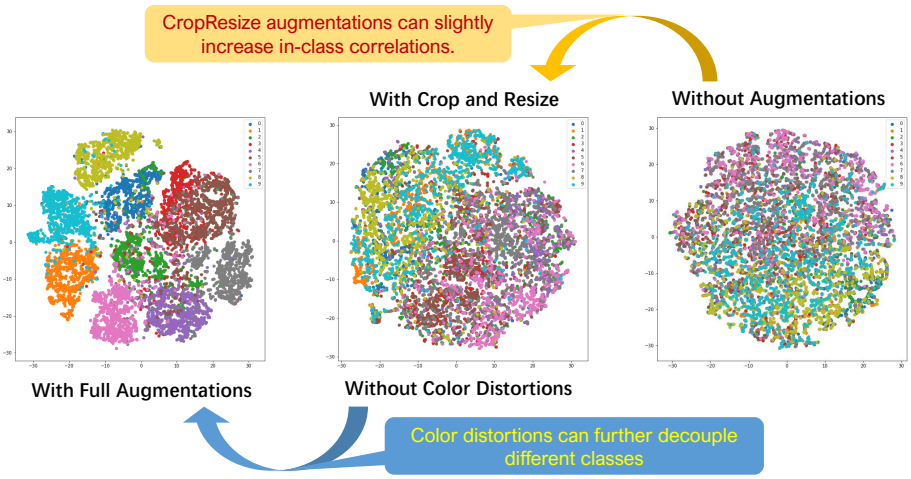


Figure 3. Evidence supporting our theoretical framework: the effects of augmentations on the learned representations of Wide-ResNet 34x5 over CIFAR10 visualized via t-SNE. The differences bewteen features learned under different augmentations shows that the neural networks will indeed learn **dense representations** if augmentation is not powerful enough.

related node and resample the others (even the latent nodes) in their generative models, reduceing the problem to supervised learning.

Theory of neural networks. There are many prior works on the supervised learning of neural networks. The works (Li & Yuan, 2017; Brutzkus & Globerson, 2017; Ge et al., 2018; Soltanolkotabi, 2017; Li et al., 2018) focus on the scenarios where data inputs are sampled from Gaussian distributions. *We consider in our paper the Gaussian part of the data to be spurious* and use augmentation to prevent learning from them. Our approach is also fundamentally different from the *neural tangent kernel* (NTK) point of view (Jacot et al., 2018; Li & Liang, 2018; Du et al., 2019; Allen-Zhu et al., 2019b;a;c; Chen et al., 2019). The NTK approach relies on approximating via first order taylor-expansion with extreme over-parameterization. More importantly, NTK cannot explain the **feature learning process** of neural networks because it is only doing linear regression over *prescribed feature map*, instead of learning the features. Some works consider the regimes beyond NTK (Allen-Zhu & Li, 2019; 2020a;b; Allen-Zhu & Li, 2020; Li et al., 2020; Bai & Lee, 2020), which shedded insights to the innerworkings of neural networks in practice.

2. Problem Setup

Notations. We use O, Ω, Θ notations to hide universal constants with respect to d and $\widetilde{O}, \widetilde{\Omega}, \widetilde{\Theta}$ notations to hide polylogarithmic factors of d. We use the notations $\mathsf{poly}(d)$, $\mathsf{polylog}(d)$ to represent constant degree polynomials of d or $\log d$. We use $[d]$ as a shorthand for the index set $\{1, \ldots, d\}$. For a matrix $\mathbf{M} \in \mathbb{R}^{d' \times d}$, we use $\mathbf{M}_j$, where $j \in [d]$, to denote its j-th column. We say an event happens with high probability (or w.h.p. for short) if the event happens with probability at least $1 - e^{-\Omega(\log^2 d)}$. We use $\mathcal{N}(\mu, \Sigma)$ to denote standard normal distribution in with mean μ and covariance matrix Σ.

2.1. Data Distribution.

We present our sparse coding model below, which form the basis of our analysis.

Definition 2.1 (sparse coding model $(\mathcal{D}_x, \mathcal{D}_z, \mathcal{D}_\xi)$)**.** We assume our raw data samples $x \in \mathbb{R}^{d_1}$ are generated i.i.d. from distribution $\mathcal{D}_x$ in the following form:

$$x = \mathbf{M}z + \xi \sim \mathcal{D}_x, \quad z \sim \mathcal{D}_z, \quad \xi \sim \mathcal{D}_\xi = \mathcal{N}(\mathbf{0}, \sigma_\xi^2 \mathbf{I}_{d_1})$$

Where $z \in \mathbb{R}^d$. We refer to z as the **sparse signal** and ξ as the **spurious dense noise**. We assume $d_1 = \mathsf{poly}(d)$ for simplicity. We have the following assumptions on $\mathbf{M}, z, \xi$ respectively:[1]

- The dictionary matrix $\mathbf{M} = [\mathbf{M}_1, \ldots, \mathbf{M}_d] \in \mathbb{R}^{d_1 \times d}$ is a column-orthonormal matrix, and satisfies $\|\mathbf{M}_j\|_\infty \le \widetilde{O}\big(\frac{1}{\sqrt{d_1}}\big)$ for all $j \in [d]$.
- The sparse latent variable $z = (z_1, \ldots, z_d)^\top \in \{-1, 0, 1\}^d$ is sampled from $\mathcal{D}_z$, we assume all z_j's are symmetric around zero, satisfying $\mathbf{Pr}(|z_j| = 1) =$

[1]The choice of $\mathbf{Pr}(|z_j| = 1) = \Theta(\frac{\log \log d}{d})$ instead of $\Theta(\frac{1}{d})$ here is to avoid the scenario where z could be zero with probability $\ge \Omega(1)$. Most of our other requirements above on the data distribution can be relaxed. Although our theory tolerates a wider range of these parameters, we choose to present the simplest setting.

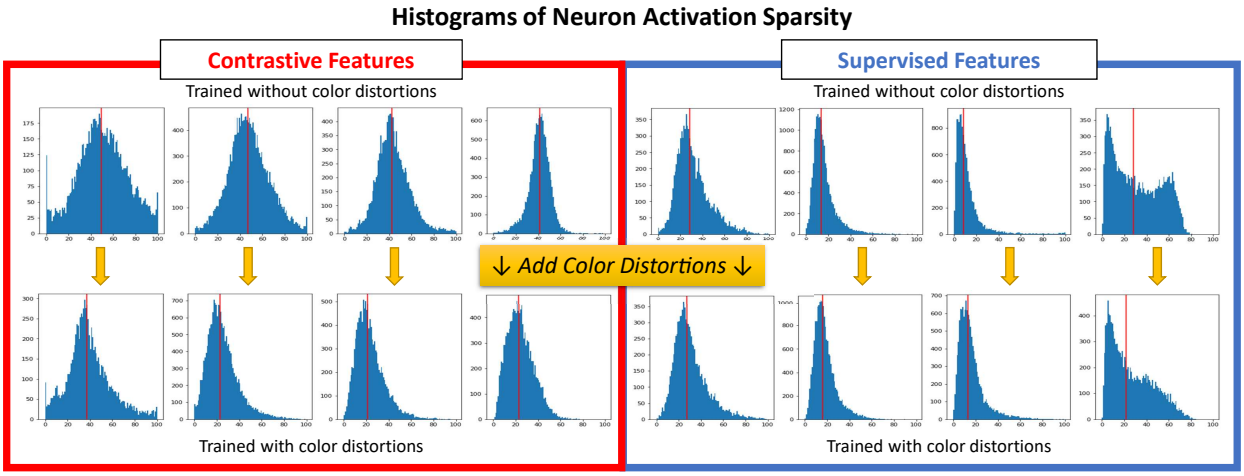


Figure 4. Another evidence supporting our theoretical framework. After adding the color distortion to augmentation, the neurons of AlexNet (2nd to 5th layer) exhibit sparser firing patterns over input images of CIFAR10. Meanwhile the networks obtained from supervised learning always have sparse activations regardless of augmentations. These observations indicate that (1). In contrastive learning, augmentations can indeed help neural nets focus on the sparse signals. (2). Sparse signals are indeed more important for the downstream tasks (such as supervised classification).

$\Theta\big(\frac{\log\log d}{d}\big)$, and are identically distributed and independent across all $j \in [d]$.

- For the spurious dense noise $\xi \sim \mathcal{N}(\mathbf{0}, \sigma_\xi^2 \mathbf{I}_{d_1})$, we assume its variance parameter $\sigma_\xi^2 = \Theta(\frac{\sqrt{\log d}}{d})$.

Why sparse coding model. Sparse coding model was first proposed by neuroscientists to model human visual systems (Olshausen & Field, 1997; 2004), where they provided experimental evidence that sparse codes can produce coding matrices for image patches that resemble known features in certain portion of the visual cortex. It has been further studied by (Földiák & Young, 1998; Vinje & Gallant, 2000; Olshausen & Field, 2004; Protter & Elad, 2009; Yang et al., 2009; Mairal et al., 2014) to model images based on the sparse occurences of objects. For the natural language data, sparse code is also found to be helpful in modelling the polysemy of words (Arora et al., 2018). Thus we believe our setting share some similar structures with practical scenarios.

Why sparse signals are more favorable. Theoretically, we argue that sparse signals are more favorable as we can see from the properties of our sparse signals $\mathbf{M}z$ and dense signals ξ:

1. **The significance of sparse signal.** Since $\sigma_\xi^2 = \Theta\big(\frac{\sqrt{\log d}}{d}\big)$, the ℓ_2-norm of ξ becomes $\|\xi\|_2^2 \geq \Omega(\mathsf{poly}(d)) \gg \|\mathbf{M}z\|_2$ w.h.p. However, whenever there is one $z_j \neq 0$, we have $|\langle \mathbf{M}z, \mathbf{M}_j \rangle| \geq \Omega(1)$ while $|\langle \xi, \mathbf{M}_j \rangle| \leq \widetilde{O}(\frac{1}{\sqrt{d}})$ with high probability. This indicates that even if the dense signal is extremely large in norm, it cannot corrupt the sparse signal.

2. **The individuality of dense signal.** For each $j \in [d]$, the sparse feature $\pm\mathbf{M}_j$ are shared by at least $\widetilde{\Omega}(\frac{1}{d})$ of the population. However, for polynomially many independent dense signal ξ_i, with high probability we have $|\left\langle \frac{\xi_i}{\|\xi_i\|_2}, \frac{\xi_j}{\|\xi_j\|_2} \right\rangle| \leq \widetilde{O}(\frac{1}{\mathsf{poly}(d)})$ for any $i \neq j$, which shows that the dense signal ξ is in some sense **"individual to each sample". This also suggests that any representations of the dense signal can hardly form separable clusters other than isolated points.**

2.2. Learner Network and Contrastive Learning Algorithm

We use a single-layer neural net $f : \mathbb{R}^{d_1} \to \mathbb{R}^m$ with ReLU activation as our contrastive learner, where m is the number of neurons. More precisely, it is defined as follows:

$$f(x) = (h_1(x), \ldots, h_m(x))^\top \in \mathbb{R}^m,$$
$$h_i(x) = \mathsf{ReLU}(\langle w_i, x \rangle - b_i) - \mathsf{ReLU}(-\langle w_i, x \rangle - b_i)$$

Such activation function h_i is a symmetrized version of ReLU activation. We initialize the parameters by $w_i^{(0)} \sim \mathcal{N}(0, \sigma_0^2 \mathbf{I}_{d_1})$ and $b_i^{(0)} = 0$, where $\sigma_0^2 = \Theta(\frac{1}{d_1 \mathsf{poly}(d)})$ is small (and also theoretically friendly). Corresponding to the two types of signals in Definition 2.1, **we call the learned weights of neural networks $\{w_i\}_{i\in[m]}$ "features", and we expand the weight of a neuron as**

$$w_i = \textstyle\sum_{j\in[d]} \langle w_i, \mathbf{M}_j \rangle \mathbf{M}_j + \sum_{j\in[d_1-d]} \langle w_i, \mathbf{M}_j^\perp \rangle \mathbf{M}_j^\perp$$

where we name the directions $\mathbf{M}_j$ and $\mathbf{M}_j^\perp$ as follows:

- We call $\mathbf{M} = [\mathbf{M}_j]_{j\in[d]}$ the **sparse features**, which is the features associated with our sparse signals $\mathbf{M}z$. These are the desired features we want our learner network to learn.
- We call $\mathbf{M}^\perp = \{\mathbf{M}_j^\perp\}_{j\in[d_1-d]}$ (the orthogonal complement of $\mathbf{M}$) the **spurious dense features**, which is only associated with our dense signal ξ. These are the undesired features for our learner.

Definition 2.2 (Contrastive loss)**.** Our contrastive loss function is based on the similarity measure defined as follows: let x and x' be two samples in $\mathbb{R}^{d_1}$, and $f : \mathbb{R}^{d_1} \to \mathbb{R}^d$ be a feature map, the similarity of the representations of x and x' is defined as

$$\mathsf{Sim}_f(x, x') := \langle f(x), \mathsf{StopGrad}(f(x'))\rangle \tag{2.1}$$

The $\mathsf{StopGrad}(\cdot)$ operator here means that we do not compute its gradient in optimization, which is inspired by recent works (Grill et al., 2020; Chen & He, 2020). Now suppose we are given a pair of positive data samples x_p, x'_p and a batch of negative data samples $\{x_{n,s}\}_{s\in\mathfrak{N}}$ (we write $\{x_{n,s}\}_{\mathfrak{N}}$ for short), let τ be the temperature parameter, the contrastive loss is defined as[2]

$$\begin{aligned}&\mathcal{L}(f, x_p, x'_p, \{x_{n,s}\}_{\mathfrak{N}})\\ &:= -\tau \log\left(\frac{e^{\mathsf{Sim}_f(x_p,x'_p)/\tau}}{\sum_{s\in\mathfrak{N}} e^{\mathsf{Sim}_f(x_p,x_{n,s})/\tau}}\right)\end{aligned} \tag{2.2}$$

However, as shown by our experiments (see Figure 2 or Figure 3), the success of contrastive learning rely on the augmentations adopted in generating the positive samples (and also the negative samples). We present our augmentation method RandomMask below, which is an analog of the random cropping data augmentation used in practice.

Definition 2.3 (RandomMask and $\mathcal{D}_{\mathbf{D}}$)**.** We first define a distribution $\mathcal{D}_{\mathbf{D}}$ over the space $\mathbb{R}^{d_1\times d_1}$ of diagonal matrices as follows: let $\mathbf{D} = \mathbf{diag}(\mathbf{D}_{\ell,\ell})_{\ell\in[d_1]} \sim \mathcal{D}_{\mathbf{D}}$ be a diagonal matrix with $\{0, 1\}$ entries, its diagonal entries $\mathbf{D}_{\ell,\ell}$ are sampled from $\mathrm{Bernoulli}\big(\frac{1}{2}\big)$ independently. Now suppose we are given a positive sample $x_p \sim \mathcal{D}_x$, we generate $\mathbf{D} \sim \mathcal{D}_{\mathbf{D}}$, and then apply the matrix $\mathbf{D}$ to generate x_p^+ and x_p^{++} as follows:

$$x_p^+ := 2\mathbf{D}x_p, \qquad x_p^{++} := 2(\mathbf{I} - \mathbf{D})x_p$$

[2]The contrastive loss (2.2) defined here have used the unnormalized representations instead of the normalized ones, which is simpler to analyze theoretically. As shown in (Chen et al., 2020a), contrastive learning using unnormalized representation can also achieve meaningful (more than 57%) ImageNet top-1 accuracy in linear evaluation of the learned representations.

Remark 2.4. We do not apply any augmentation to our negative samples in order for simplicity of theory. And also we point out that adding such augmentations do not reveal any further insights, since we do not expect the augmentation to decouple any correlations other than that between positive samples. Nevertheless our theory can easily adapt to the setting where augmentations are applied to every input data.

Intuitions behind the RandomMask **augmentation.** Intuitively, the RandomMask data augmentation simply masks out roughly a half of the coordinates in the data. The contrastive learning objective asks to learn features that can match *two disjoint set of the coordinates* of given data points. Suppose we can maintain the correlations of desired signals between the disjoint coordinates and remove the undesired correlations, then we can force the algorithm to learn from the desired signals. We will discuss the effects of augmentations with more detail in Section 4.

Significance of our analysis on the data augmentations. Our analysis on the data augmentation are fundamentally different from those in (Tsai et al., 2020; Tian et al., 2020b; Wei et al., 2020; Lee et al., 2020). In (Tsai et al., 2020; Tian et al., 2020b), they argued their data augmentations can change the latent variables unretaled to the downstream tasks, while real-life augmentations can only affect the observables, and cannot identify which latents are the task-specific ones. (Wei et al., 2020) assumed their augmentations are only picking data points inside a small neighborhood of the original data (in the observable space), which is also untrue in practice. Indeed, common augmentations like crop-resize and color distortions can considerably change the data, making it very distant to the original data in the observable space. Our analysis of RandomMask makes a step toward understanding realistic data augmentations in deep learning.

Training algorithm using SGD. We consider two cases: training with augmentation and without augmentation:

- **With Augmentation.** We perform stochastic gradient descent on the following objectives: for each training iterations $t \geq 0$ and the contrastive learner f_t, the objectives is defined as follows:

$$\begin{aligned}\mathbf{Obj}(f_t) &:= L(f_t) + \lambda \sum_{i\in[m]} \|w_i^{(t)}\|_2^2,\\ L(f_t) &:= \mathop{\mathbb{E}}_{x_p^+, x_p^{++}, \mathfrak{N}} \left[\mathcal{L}(f_t, x_p^+, x_p^{++}, \mathfrak{N})\right]\end{aligned}$$

  where $\lambda \in [\frac{1}{d^{1.001}}, \frac{1}{d^{1.499}}]$ is the regularization parameter, $L(f_t)$ is the population loss and $x_p, \{x_{n,s}\}_{\mathfrak{N}}$ are sampled from $\mathcal{D}_x$, x_p^+, x_p^{++} are obtained by applying RandomMask to x_p. At each iteration t, let $\eta = \frac{1}{\mathsf{poly}(d)}$ be the learning rate, we update as:

$$w_i^{(t+1)} \leftarrow w_i^{(t)} - \eta\nabla_{w_i}\mathbf{Obj}(f_t)$$

- **Without Augmentation.** We perform stochastic gradient descent on the following modified objectives $\mathbf{Obj}_{\mathsf{NA}}(f_t)$:

$$\mathbf{Obj}_{\mathsf{NA}}(f_t) := L_{\mathsf{NA}}(f_t) + \lambda \sum_{i\in[m]} \|w_i^{(t)}\|_2^2,$$
$$L_{\mathsf{NA}}(f_t) := \mathop{\mathbb{E}}_{x_p,\mathfrak{N}}[\mathcal{L}(f_t, x_p, x_p, \mathfrak{N})]$$

  where $\lambda \le O(1/d)$ can be arbitrary. The learning rate $\eta \le o(1)$ can also be arbitrary. We update as:

$$w_i^{(t+1)} \leftarrow w_i^{(t)} - \eta \nabla_{w_i} \mathbf{Obj}_{\mathsf{NA}}(f_t)$$

We manually tune bias[3] $b_i^{(t)}$ during the training process as follows: let $T_1 = \Theta\left(\frac{d \log d_1}{\eta \log\log d}\right)$ be the iteration when all $\|w_i^{(0)}\|_2 \le \gamma c_0 \|w_i^{(t)}\|_2$. At $t = T_1$, we reset the bias $b_i^{(t)} = \sqrt{\frac{2\log d}{d}}\|w_i^{(t)}\|_2$ and update by $b_i^{(t+1)} = b_i^{(t)}(1+\eta_{b,t})$, where $\eta_{b,t} = \max\{\frac{\eta}{d}, \frac{\|w_i^{(t+1)}\|_2}{\|w_i^{(t)}\|_2} - 1\}$ if $b_i^{(t)} \le \frac{\mathsf{polylog}(d)}{\sqrt{d}}$.

3. Main Results

We now state the main theorems of this paper in our setting. We argue that contrastive learning objective learns completely different features with/without data augmentation. **Moreover, to further illustrate the how these learned features are different with/without data augmentation, we also consider two simple downstream tasks to evaluate the performance of contrastive learning**. We argue that using a linear function taking the learned representation as input to perform these tasks can be more efficient than using raw inputs, it should be considered as successful representation learning.

Definition 3.1 (downstream tasks)**.** We consider two simple supervised tasks, regression and classification, based on the label functions defined below:

- Regression: For each $x = \mathbf{M}z + \xi \sim \mathcal{D}_x$, we define its label $y = \langle w^\star, z\rangle$, where $w^\star \in \mathbb{R}^d$.
- Classification: For each $x = \mathbf{M}z + \xi \sim \mathcal{D}_x$, we define $y = \mathrm{sign}(\langle w^\star, z\rangle)$, where $w^\star \in \mathbb{R}^d$.

where in both cases we assume $w^\star$ satisfies $|w_j^\star| = \Theta(1)$ for all $j \in [d]$.

Given these downstream tasks, our goal of representations learning is to obtain suitable feature representations and train a linear classifier over them. Specifically, let $f(\cdot)$ be the obtained representation map, we use optimization tool[4] to find w^* such that

$$w^* = \arg\min_{w\in\mathbb{R}^m} \mathbb{E}[\widetilde{\mathcal{L}}(w^\top f(x), y)]$$

where $\widetilde{\mathcal{L}}(\cdot,\cdot)$ is the loss function for the downstream tasks considered: For regression, it is the ℓ_2 loss $\widetilde{\mathcal{L}}(\hat{y}, y) = (\hat{y} - y)^2$; For classification, it is the logistic loss $\widetilde{\mathcal{L}}(\hat{y}, y) = \log(1 + e^{-\hat{y}y})$. It should be noted that these tasks can be done efficiently by neural networks via supervised learning as shown by Allen-Zhu & Li (2020b). However, such task cannot be done by doing linear regression over the input x, since even if one can locate the desired features $\mathbf{M}$, the noise level $\sigma_\xi^2 = \Theta(\frac{\sqrt{\log d}}{d})$ is still much larger than the signal size $\mathbb{E}[z_j^2] = \Theta(\frac{\log\log d}{d})$, thus linear models will fail with constant probability.

3.1. Contrastive Learning Without Augmentations

We present our theorem for the learned features without using any augmentations.

Theorem 3.2 (Contrastive features learned without augmentation)**.** *Let f_t^{NA} be the neural network trained by conrtastive learning without any data augmentations, and using $|\mathfrak{N}| = \mathsf{poly}(d)$ many negative samples, we have objective guarantees $L_{\mathsf{NA}}(f_t^{\mathsf{NA}}) = o(1)$ for any $t \ge \frac{\mathsf{poly}(d)}{\eta}$. Moreover, given a data sample $x = \mathbf{M}z + \xi \sim \mathcal{D}_x$, with high probability it holds:*

$$\left\langle \frac{f_t^{\mathsf{NA}}(x)}{\|f_t^{\mathsf{NA}}(x)\|_2}, \frac{f_t^{\mathsf{NA}}(\xi)}{\|f_t^{\mathsf{NA}}(\xi)\|_2} \right\rangle \ge 1 - \widetilde{O}\left(\frac{1}{\mathsf{poly}(d)}\right)$$

This results means that in the representations of f_t, the sparse signal $\mathbf{M}z$ are completely overwhelmed by the spurious dense signal ξ. It would be easy to verify the following corollary:

Corollary 3.3 (Downstream task performance)**.** *The learned network f_t^{NA}, where $t \ge 0$, fail to achieve meaningful ℓ_2-loss/accuracy in the downstream tasks in Definition 3.1. More specifically, no matter how many labeled data we have for downstream linear evaluation (where f_t^{NA} is frozen):*

- *For regression, we have*

$$\mathop{\mathbb{E}}_{x\sim\mathcal{D}_x} |y - \langle w^*, f_t^{\mathsf{NA}}\rangle(x)|^2 \ge \Omega(1)$$

- *For classification, we have*

$$\Pr_{x\sim\mathcal{D}_x}[y = \mathrm{sign}(\langle w^*, f_t^{\mathsf{NA}}(x)\rangle)] = o(1)$$

[3] In fact, when trained without augmentations, the biases can be tuned arbitrarily as long as the neurons are not killed. It will not affect our results.

[4] Since the downstream learning tasks only involve linear learners on convex objectives, for simplicity, we directly argue the property of the minimizer for these downstream training objectives.

3.2. Contrastive Learning With Augmentation

We present our results of the learned features after successful training with augmentations.

Theorem 3.4 (Contrastive features learned with augmentation)**.** *Let $m = d^{1.01}$ be the number of neurons, $\tau = \mathsf{polylog}(d)$, and $|\mathfrak{N}| = \mathsf{poly}(d)$ be the number of negative samples. Suppose we train the neural net f_t via contrastive learning with augmentation, then for some iterations $T \in [T_3, T_4]$, where $T_3 = \frac{d^{1.01}}{\eta}, T_4 = \frac{d^{1.99}}{\eta}$, we have objective guarantees*

$$\frac{1}{T}\sum_{T_3\le t<T} \mathbf{Obj}(f_t) \le o(1), \quad \frac{1}{T}\sum_{T_3\le t<T} L(f_t) \le o(1)$$

Moreover, for each neuron $i \in [m]$ and $t \in [T_3, T_4]$, contrastive learning will learn the following set of features:

$$w_i^{(t)} = \sum_{j\in\mathcal{N}_i} \alpha_{i,j}\mathbf{M}_j + \sum_{j\notin\mathcal{N}_i} \alpha'_{i,j}\mathbf{M}_j + \sum_{j\in[d_1]\setminus[d]} \beta_{i,j}\mathbf{M}_j^{\perp}$$

where $\alpha_{i,j} \in [\frac{\tau}{d^c}, \tau]$, $|\mathcal{N}_i| = O(1)$, $\alpha'_{i,j} \le o(\frac{1}{\sqrt{d}})\|w_i^{(t)}\|_2$ and $|\beta_{i,j}| \le o(\frac{1}{\sqrt{d_1}})\|w_i^{(t)}\|_2$, for some small constant $c < \frac{1}{1000}$. Furthermore, for each dictionary atom $\mathbf{M}_j$, there are at most $o(m/d)$ many $i \in [m]$ such that $j \in \mathcal{N}_i$, and at least $\Omega(1)$ many $i \in [m]$ such that $\mathcal{N}_i = \{j\}$.

This result indicates the following: let $x = \mathbf{M}z + \xi \sim \mathcal{D}_x$ be a data sample and $f_t, t \in [T_3, T_4]$ be the trained network, then $\|f_t(x) - f_t(\mathbf{M}z)\|_2 \le \widetilde{O}(\frac{1}{\sqrt{d}})$ with high probability, while $\|f(\mathbf{M}z)\|_2 \ge \Omega(1)$ with probability at least $1 - \frac{1}{\mathsf{polylog}(d)}$. Thus the learned feature map has successfully removed the spurious dense noise ξ from the model/representation. We have a direct corollary following this theorem.

Corollary 3.5 (Downstream task performance)**.** *The learned feature map f_t, $t \in [\frac{d^{1.01}}{\eta}, \frac{d^{1.49}}{\eta}]$ obtained by contrastive learning perform well in all the downstream tasks defined in Definition 3.1. Specifically, we have*

1. *For the regression task, with sample complexity at most $\widetilde{O}(d)$, we can obtain $w^* \in \mathbb{R}^m$ such that*
$$\mathop{\mathbb{E}}_{x\sim\mathcal{D}_x} |y - \langle w^*, f_t(x)\rangle|^2 = o(1)$$
2. *For the classification task, again by using logistic regression over feature map f_t, with sample complexity at most $\widetilde{O}(d)$, we can find $w^* \in \mathbb{R}^d$ such that*
$$\mathop{\mathbf{Pr}}_{x\sim\mathcal{D}_x}[y = \mathrm{sign}(\langle w^*, f_t(x)\rangle)] = 1 - o(1)$$

4. Proof Intuition: The Feature Decoupling Principle

Theoretically speaking, contrastive learning objectives can be view as two parts, as was also argued in Wang & Isola (2020):

$$\mathcal{L} = -\mathsf{Sim}_f(x_p, x'_p) + \tau \log\left(\sum_{x\in\mathfrak{B}} e^{\mathsf{Sim}_f(x_p,x)/\tau}\right)$$

where the first part $-\mathsf{Sim}_f(x_p, x'_p)$ emphasize similarity between positive samples, and the second part $\tau \log\left\{\sum_{x\in\mathfrak{B}} e^{\mathsf{Sim}_f(x_p,x)/\tau}\right\}$ emphasize dissimilarities between the positive and negative samples. To understand what happens in the learning process, we separately discuss the cases of learning with/without augmentations below:

Why does contrastive learning prefer spurious dense noise without augmentation? Without data augmentation, we simply have $x_p = x'_p$. In this case, contrastive learning will learn to emphasize the signals that simultaneously maximize the correlation $\langle f(x_p^{++}), f(x_p^{+})\rangle = \|f_t(x_p)\|_2^2$ and minimize $\langle f(x_{n,s}), f(x_p^{+})\rangle$ by learning from all the available signals. **However, in our sparse coding model $x = \mathbf{M}z + \xi$, the spurious dense features ξ has much larger ℓ_2-norm and the least correlations between different samples** (see Section 2 for discussion). In contrast, the sparse signals $\mathbf{M}z = \sum_j \mathbf{M}_j z_j$ display larger correlations between different samples because of possible co-occurences of features $\mathbf{M}_j$ (i.e., at least $\widetilde{\Omega}(\frac{1}{d})$ portion of the data contain feature $\mathbf{M}_j$). Thus the our contrastive learner will focus on learning the features associated with the dense noise ξ, and fail to emphasize sparse features.

Feature Decoupling: How does augmentation remove the spurious dense noise: Theoretically, we show how data augmentations help contrastive learning, which demonstrate the principle of **feature decoupling**.

Specifically, under our data model defined in Definition 2.1, if no augmentations are applied to the two positive samples x_p^+, x_p^{++} generated from $x_p = \mathbf{M}z_p + \xi_p \sim \mathcal{D}_x$, their correlations will mostly come from the inner product of noise $\langle \xi_p, \xi_p\rangle$, which can easily overwhelm those from the sparse signals $\langle \mathbf{M}z_p, \mathbf{M}z_p\rangle$. Nevertheless, we have a simple observation: each coordinate ξ_j are independent, which enables a simple method to decorrelate the positive samples – By randomly applying two completely opposite masks $\mathbf{D}$ and $\mathbf{I} - \mathbf{D}$ to the data x to generate two positive samples x_p^+ and x_p^{++}. From our observation, such augmentations can make the dense signals $\mathbf{D}\xi$ and $(\mathbf{I} - \mathbf{D})\xi$ of x_p^+ and x_p^{++} independent to each other. This independence will decouple the dense features between positive samples, which substantially reduces the gradients of the dense features.

However, the sparse signals are more resistant to data augmentation. As long as the sparse signals $\mathbf{M}z = \sum_{j\in[d]} \mathbf{M}_j z_j$ span across the space, they will show up in both x_p^+ and x_p^{++}, so that their correlations will remain in the representations. More precisely, whenever a sparse signal $\mathbf{M}_j$ is present (meaning its latent variable

$z_j \neq 0$), it can be recovered both from $2\mathbf{D}\mathbf{M}z$ and from $2(\mathbf{I}-\mathbf{D})\mathbf{M}z$ with the correct decoding: e.g. we have $T_b(\langle \mathbf{M}_j, 2\mathbf{D}x\rangle) \approx T_b(\langle \mathbf{M}_j, 2(\mathbf{I}-\mathbf{D})x\rangle) \approx z_j$, where $T_b(x) = x\mathbb{1}_{|x|\geq b}$ is a threshold operator with a proper bias $b > 0$. Unless in very rare case $\mathbf{M}_j$ is completely masked by augmentations (that is $\mathbf{D}\mathbf{M}_j = 0$ or $(\mathbf{I}-\mathbf{D})\mathbf{M}_j = 0$), the sparse signals will remain their correlations in the feature representations, which will be reinforced by neural networks following the SGD trajectory.

5. Conclusion and Discussion

In this work, we show a theoretical result toward understanding how contrastive learning method learns the feature representations in deep learning. We present the feature decoupling principle to tentatively explain how augmentations work in contrastive learning. We also provide empirical evidence supporting our theory, which suggest that augmentations are necessary if we want to learn the desired features and remove the undesired ones. We hope our theory could shed light on the innerworkings of how neural networks perform representation learning in self-supervised setting.

However, we also believe that our results can be significantly improved if we can build on more realistic data distributions. For example, real life image data should be more suitably modeled as "hierachical sparse coding model" instead of the current simple linear sparse coding model. We believe that deeper network would be needed in the new model. Studying contrastive learning over those data models and deep networks is an important open direction.

Acknowledgements

We would like to thank Zeyuan Allen-Zhu for many helpful suggestions on the experiments.

References

Allen-Zhu, Z. and Li, Y. What can resnet learn efficiently, going beyond kernels? In *NeurIPS 2019 : Thirty-third Conference on Neural Information Processing Systems*, pp. 9017–9028, 2019.

Allen-Zhu, Z. and Li, Y. Backward feature correction: How deep learning performs deep learning. *arXiv preprint arXiv:2001.04413*, 2020a.

Allen-Zhu, Z. and Li, Y. Feature purification: How adversarial training performs robust deep learning. *arXiv preprint arXiv:2005.10190*, 2020b.

Allen-Zhu, Z. and Li, Y. Towards understanding ensemble, knowledge distillation and self-distillation in deep learning. *arXiv preprint arXiv:2012.09816*, 2020.

Allen-Zhu, Z., Li, Y., and Liang, Y. Learning and generalization in overparameterized neural networks, going beyond two layers. In *NeurIPS 2019 : Thirty-third Conference on Neural Information Processing Systems*, pp. 6158–6169, 2019a.

Allen-Zhu, Z., Li, Y., and Song, Z. A convergence theory for deep learning via over-parameterization. In *ICML 2019 : Thirty-sixth International Conference on Machine Learning*, pp. 242–252, 2019b.

Allen-Zhu, Z., Li, Y., and Song, Z. On the convergence rate of training recurrent neural networks. In *Advances in Neural Information Processing Systems*, pp. 6676–6688, 2019c.

Arjovsky, M., Chintala, S., and Bottou, L. Wasserstein generative adversarial networks. In *International Conference on Machine Learning*, pp. 214–223, 2017.

Arora, S., Li, Y., Liang, Y., Ma, T., and Risteski, A. Linear algebraic structure of word senses, with applications to polysemy. *Transactions of the Association for Computational Linguistics*, 6:483–495, 2018.

Arora, S., Khandeparkar, H., Khodak, M., Plevrakis, O., and Saunshi, N. A theoretical analysis of contrastive unsupervised representation learning. *arXiv preprint arXiv:1902.09229*, 2019.

Bai, Y. and Lee, J. D. Beyond Linearization: On Quadratic and Higher-Order Approximation of Wide Neural Networks. *arXiv:1910.01619 [cs, math, stat]*, February 2020.

Brutzkus, A. and Globerson, A. Globally optimal gradient descent for a convnet with gaussian inputs. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pp. 605–614. JMLR. org, 2017.

Carreira-Perpiñán, M. Á. and Hinton, G. E. On contrastive divergence learning. In *AISTATS*, 2005.

Chen, S., Dobriban, E., and Lee, J. H. A group-theoretic framework for data augmentation. *Journal of Machine Learning Research*, 21(245):1–71, 2020a.

Chen, T., Kornblith, S., Norouzi, M., and Hinton, G. A simple framework for contrastive learning of visual representations. In *ICML 2020: 37th International Conference on Machine Learning*, volume 1, pp. 1597–1607, 2020b.

Chen, T., Kornblith, S., Swersky, K., Norouzi, M., and Hinton, G. Big Self-Supervised Models are Strong Semi-Supervised Learners. *Arxiv*, pp. 13, 2020.

Chen, X. and He, K. Exploring Simple Siamese Representation Learning. *arXiv:2011.10566 [cs]*, November 2020.

Chen, Z., Cao, Y., Zou, D., and Gu, Q. How much over-parameterization is sufficient to learn deep relu networks? *arXiv preprint arXiv:1911.12360*, 2019.

Coates, A., Ng, A. Y., and Lee, H. An analysis of single-layer networks in unsupervised feature learning. In *AISTATS*, 2011.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACL-HLT*, pp. 4171–4186, 2019.

Du, S. S., Lee, J. D., Li, H., Wang, L., and Zhai, X. Gradient descent finds global minima of deep neural networks. In *36th International Conference on Machine Learning, ICML 2019*, pp. 1675–1685, 2019.

Földiák, P. and Young, M. P. *Sparse coding in the primate cortex*. 1998.

Ge, R., Lee, J. D., and Ma, T. Learning one-hidden-layer neural networks with landscape design. In *6th International Conference on Learning Representations, ICLR 2018*, 2018.

Grill, J.-B., Strub, F., Altché, F., Tallec, C., Richemond, P. H., Buchatskaya, E., Doersch, C., Pires, B. A., Guo, Z. D., Azar, M. G., Piot, B., Kavukcuoglu, K., Munos, R., and Valko, M. Bootstrap Your Own Latent A New Approach to Self-Supervised Learning. *Ar*, pp. 14, 2020.

Gutmann, M. U. and Hyvärinen, A. Noise-contrastive estimation of unnormalized statistical models, with applications to natural image statistics. *Journal of Machine Learning Research*, 13(Feb):307–361, 2012.

He, K., Fan, H., Wu, Y., Xie, S., and Girshick, R. Momentum contrast for unsupervised visual representation learning. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9729–9738, 2020.

Jacot, A., Gabriel, F., and Hongler, C. Neural tangent kernel: Convergence and generalization in neural networks. In *Advances in neural information processing systems*, pp. 8571–8580, 2018.

Jing, L. and Tian, Y. Self-supervised visual feature learning with deep neural networks: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1–1, 2020.

Kim, M., Tack, J., and Hwang, S. J. Adversarial self-supervised contrastive learning. In *Advances in Neural Information Processing Systems*, volume 33, 2020.

Kingma, D. P. and Welling, M. Auto-encoding variational bayes. In *ICLR 2014 : International Conference on Learning Representations (ICLR) 2014*, 2014.

Lee, J. D., Lei, Q., Saunshi, N., and Zhuo, J. Predicting What You Already Know Helps: Provable Self-Supervised Learning. *arXiv:2008.01064 [cs, stat]*, August 2020.

Li, Y. and Liang, Y. Learning overparameterized neural networks via stochastic gradient descent on structured data. In *Advances in Neural Information Processing Systems*, pp. 8157–8166, 2018.

Li, Y. and Yuan, Y. Convergence analysis of two-layer neural networks with relu activation. In *Advances in neural information processing systems*, pp. 597–607, 2017.

Li, Y., Ma, T., and Zhang, H. Algorithmic regularization in over-parameterized matrix sensing and neural networks with quadratic activations. In *COLT 2018: 31st Annual Conference on Learning Theory*, pp. 2–47, 2018.

Li, Y., Ma, T., and Zhang, H. R. Learning over-parametrized two-layer relu neural networks beyond ntk. In *COLT*, pp. 2613–2682, 2020.

Mairal, J., Bach, F., and Ponce, J. *Sparse Modeling for Image and Vision Processing*. 2014.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.

Nguyen, T. V., Wong, R. K. W., and Hegde, C. Benefits of jointly training autoencoders: An improved neural tangent kernel analysis. *arXiv preprint arXiv:1911.11983*, 2019.

Olshausen, B. A. and Field, D. J. Sparse coding with an overcomplete basis set: A strategy employed by v1 ? *Vision Research*, 37(23):3311–3325, 1997.

Olshausen, B. A. and Field, D. J. Sparse coding of sensory inputs. *Current Opinion in Neurobiology*, 14(4):481–487, 2004.

Oord, A. v. d., Li, Y., and Vinyals, O. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

Protter, M. and Elad, M. Image sequence denoising via sparse and redundant representations. *IEEE Transactions on Image Processing*, 18(1):27–35, 2009.

Radford, A., Metz, L., and Chintala, S. Unsupervised representation learning with deep convolutional generative adversarial networks. In *ICLR 2016 : International Conference on Learning Representations 2016*, 2016.

Radhakrishnan, A., Yang, K., Belkin, M., and Uhler, C. Memorization in overparameterized autoencoders. *arXiv preprint arXiv:1810.10333*, 2018.

Smith, N. A. and Eisner, J. Contrastive estimation: Training log-linear models on unlabeled data. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, ACL '05, pp. 354–362, USA, 2005. Association for Computational Linguistics. doi: 10.3115/1219840.1219884.

Soltanolkotabi, M. Learning relus via gradient descent. In *Advances in Neural Information Processing Systems*, volume 30, pp. 2007–2017, 2017.

Sutskever, I., Vinyals, O., and Le, Q. V. Sequence to Sequence Learning with Neural Networks. In *Advances in Neural Information Processing Systems 27*, volume 27, pp. 3104–3112, 2014.

Tian, Y., Krishnan, D., and Isola, P. Contrastive multiview coding. In *ECCV (11)*, pp. 776–794, 2019.

Tian, Y., Sun, C., Poole, B., Krishnan, D., Schmid, C., and Isola, P. What makes for good views for contrastive learning. In *Advances in Neural Information Processing Systems*, volume 33, 2020a.

Tian, Y., Yu, L., Chen, X., and Ganguli, S. Understanding self-supervised learning with dual deep networks. *arXiv preprint arXiv:2010.00578*, 2020b.

Tosh, C., Krishnamurthy, A., and Hsu, D. Contrastive estimation reveals topic posterior information to linear models. *arXiv preprint arXiv:2003.02234*, 2020.

Tosh, C., Krishnamurthy, A., and Hsu, D. Contrastive learning, multi-view redundancy, and linear models. In *Algorithmic Learning Theory*, pp. 1179–1206. PMLR, 2021.

Tsai, Y.-H. H., Wu, Y., Salakhutdinov, R., and Morency, L.-P. Demystifying self-supervised learning: An information-theoretical framework. 2020.

Vinje, W. E. and Gallant, J. L. Sparse coding and decorrelation in primary visual cortex during natural vision. *Science*, 287(5456):1273–1276, 2000.

Wang, T. and Isola, P. Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In *ICML*, pp. 9929–9939, 2020.

Wei, C., Shen, K., Chen, Y., and Ma, T. Theoretical Analysis of Self-Training with Deep Networks on Unlabeled Data. *arXiv:2010.03622 [cs, stat]*, October 2020.

Yang, J., Yu, K., Gong, Y., and Huang, T. Linear spatial pyramid matching using sparse coding for image classification. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1794–1801, 2009.