CLOCS: CONTRASTIVE LEARNING OF CARDIAC SIGNALS ACROSS SPACE, TIME, AND PATIENTS

Anonymous authors
Paper under double-blind review

ABSTRACT

The healthcare industry generates troves of unlabelled physiological data. This data can be exploited via contrastive learning, a self-supervised pre-training method that encourages representations of instances to be similar to one another. We propose a family of contrastive learning methods, CLOCS, that encourages representations across space, time, *and* patients to be similar to one another. We show that CLOCS consistently outperforms the state-of-the-art methods, BYOL and SimCLR, when performing a linear evaluation of, and fine-tuning on, downstream tasks. We also show that CLOCS achieves strong generalization performance with only 25% of labelled training data. Furthermore, our training procedure naturally generates patient-specific representations that can be used to quantify patient-similarity.

1 INTRODUCTION

At present, the healthcare system is unable to sufficiently leverage the large, unlabelled datasets that it generates on a daily basis. This is partially due to the dependence of deep learning algorithms on high quality labels for good generalization performance. However, arriving at such high quality labels in a clinical setting where physicians are squeezed for time and attention is increasingly difficult. To overcome such an obstacle, self-supervised techniques have emerged as promising methods. These methods exploit the unlabelled dataset to formulate pretext tasks such as predicting the rotation of images (Gidaris et al., 2018), their corresponding colourmap (Larsson et al., 2017), and the arrow of time (Wei et al., 2018). More recently, contrastive learning was introduced as a way to learn representations of instances that share some context. By capturing this high-level shared context (e.g., medical diagnosis), representations become invariant to the differences (e.g., input modalities) between the instances.

Contrastive learning can be characterized by three main components: 1) a positive and negative set of examples, 2) a set of transformation operators, and 3) a variant of the noise contrastive estimation loss. Most research in this domain has focused on curating a positive set of examples by exploiting data temporality (Oord et al., 2018), data augmentations (Chen et al., 2020), and multiple views of the same data instance (Tian et al., 2019). These methods are predominantly catered to the image-domain and central to their implementation is the notion that shared context arises from the same instance. We believe this precludes their applicability to the medical domain where physiological time-series are plentiful. Moreover, their interpretation of shared context is limited to data from a common source where that source is the individual data instance. In medicine, however, shared context can occur at a higher level, the patient level. This idea is central to our contributions and will encourage the development of representations that are patient-specific. Such representations have the potential to be used in tasks that exploit patient similarity such as disease subgroup clustering and discovery. As a result of the process, medical practitioners may receive more interpretable outputs from networks.

In this work, we leverage electrocardiogram (ECG) signals to learn patient-specific representations in a self-supervised manner via contrastive learning. To do so, we exploit the fact that ECG signals summarize both temporal and spatial information. The latter can be understood in terms of projections of the same electrical signal onto multiple axes, also known as leads.

Contributions. Our contributions are the following:

1. **Contrastive Multi-segment Coding** - we encourage representations of pairs of temporal segments of data from the same patient to be similar to one another.

2. **Contrastive Multi-lead Coding** - we encourage representations of multiple spatial projections (ECG leads) of the same temporal segment from the same patient to be similar to one another.
3. **Contrastive Multi-segment and Multi-lead Coding** - we encourage representations of both temporal segments and multiple projections from the same patient to be similar to one another.
4. **Patient-specific Noise Contrastive Estimation Loss** - we adapt the loss introduced by Tian et al. (2019) to account for patient-specific instances.

2 Related Work

Contrastive Learning. In contrastive predictive coding, Oord et al. (2018) use representations of current segments to predict those of future segments. More recently, Tian et al. (2019) propose contrastive multi-view coding where multiple views of the same image are treated as 'shared context'. He et al. (2019); Chen et al. (2020); Grill et al. (2020) exploit the idea of instance discrimination (Wu et al., 2018) and interpret multiple views as stochastically augmented forms of the same instance. They explore the benefit of sequential data augmentations and show that cropping and colour distortions are the most important. These augmentations, however, do not trivially extend to the time-series domain. Shen et al. (2020) propose to create mixtures of images to smoothen the output distribution and thus prevent the model from being overly confident. Time Contrastive Learning (Hyvarinen & Morioka, 2016) performs contrastive learning over temporal segments in a signal and illustrate the relationship between their approach and ICA. In contrast to our work, they formulate their task as prediction of the segment index within a signal and perform limited experiments that do not exploit the noise contrastive estimation (NCE) loss. Bachman et al. (2019) Time Contrastive Networks (Sermanet et al., 2017) attempt to learn commonalities across views and differences across time. In contrast, our work focuses on identifying commonalities across *both* spatial and temporal components of data.

Self-Supervision for Medical Time-Series. Miotto et al. (2016) propose DeepPatient, a 3-layer stacked denoising autoencoder that attempts to learn a patient representation using electronic health record (EHR) data. Although performed on a large proprietary dataset, their approach is focused on EHRs and does not explore contrastive learning for physiological signals. Sarkar & Etemad (2020) apply existing self-supervised methods on ECG recordings in the context of affective computing. The methods implemented include defining pretext classification tasks such as temporal inversion, negation, time-warping, etc. Their work is limited to affective computing, does not explore contrastive learning, and does not exploit multi-lead data as we do. Lyu et al. (2018) explore a sequence to sequence model to learn representations from EHR data in the eICU dataset. In the process, they minimize the reconstruction error of the input time-series. Li et al. (2020) leverage the aforementioned unsupervised learning technique on a large clinical dataset, CPRD, to obtain uncertainty estimates for predictions. To the best of our knowledge, we are the first to design a contrastive learning approach for physiological time-series.

3 Background

3.1 Contrastive Learning

Assume the presence of a learner $f_\theta : x \in \mathbb{R}^D \rightarrow h \in \mathbb{R}^E$, parameterized by θ, which maps a D-dimensional input, x, to an E-dimensional representation, h. Further assume the presence of an unlabelled dataset, $X \in \mathbb{R}^{N \text{x} D}$, where N is the total number of instances.

Each unlabelled instance, $x^i \in X$, is exposed to a set of transformations, T_A and T_B, such that $x^i_A = T_A(x^i)$ and $x^i_B = T_B(x^i)$. Such transformations can consist of two different data augmentation procedures such as random cropping and flipping. These transformed instances now belong to an augmented dataset, $X' \in \mathbb{R}^{N \text{x} D \text{x} V}$, where V is equal to the number of applied transformations. In contrastive learning, representations, $h^i_A = f_\theta(x^i_A)$ and $h^i_B = f_\theta(x^i_B)$, are said to share context and constitute a positive pair because they are derived from the same original instance, x^i. Representations within a positive pair are encouraged to be similar to one another and dissimilar to representations of all other instances, $h^j_A, h^j_B \; \forall j \; j \neq i$. The similarity of these representations, $s(h^i_A, h^i_B)$, is quantified via a metric, s, such as cosine similarity. By encouraging high similarity between representations in the positive pair, the goal is to learn representations that are invariant to different transformations of the same instance.

4 METHODS

4.1 POSITIVE AND NEGATIVE PAIRS OF REPRESENTATIONS

Representations that are derived from the same *instance* are typically assumed to share context. This approach, however, fails to capture commonalities present across instances. In the medical domain, for example, multiple recordings from the same patient may share context. Inherent to our approach is the redefinition of a positive pair to refer to representations of transformed instances that belong to the same *patient*. We outline how to arrive at these transformed instances next.

4.2 TRANSFORMATION OPERATORS

When choosing the transformation operators, T, that are applied to each instance, the principal desideratum is that they must capture invariances in the ECG recording. In designing transformation operators, we exploit exploit both temporal and spatial information available within an ECG recording from a particular patient.

Contrastive Multi-segment Coding (CMSC). Given an ECG recording, x^i, with duration S seconds, we can extract V non-overlapping temporal segments, each with duration S/V seconds. If $V = 2$, for example, $x^i_{t1} = T_{t1}(x^i)$ and $x^i_{t2} = T_{t2}(x^i)$ where t indicates the timestamp of the temporal segment. In Fig. 1 (left), we show this pair of transformed ECG recordings (we defer an in-depth description till the end of this section). To exploit temporal invariances in the ECG, we define representations of these adjacent and non-overlapping temporal segments to be positive pairs.

Contrastive Multi-lead Coding (CMLC). ECG recordings can be thought of as different projections of the same electrical signal, also known as leads, L. For example, in the presence of two leads, $L1$ and $L2$, then $x^i_{L1} = T_{L1}(x^i)$ and $x^i_{L2} = T_{L2}(x^i)$. In Fig. 1 (center), we show this pair of transformed ECG recordings. To exploit spatial invariances in the ECG, we define representations of these different projections of the same temporal segment to be positive pairs.

Contrastive Multi-segment Multi-lead Coding (CMSMLC). To simultaneously exploit both temporal and spatial invariances in the ECG, we define representations of non-overlapping temporal

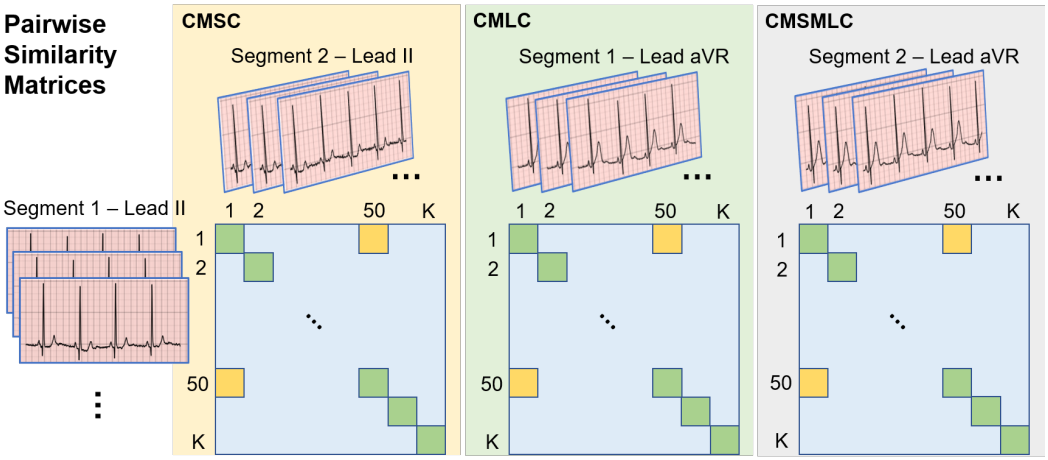


Figure 1: Pairwise similarity matrices for a mini-batch with K instances used in (left) **Contrastive Multi-segment Coding**, (centre) **Contrastive Multi-lead Coding**, and (right) **Contrastive Multi-segment Multi-lead Coding**. Additional matrices would be generated based on all pairs of applied transformation operators, T_A and T_B. Exemplar transformed ECG instances are illustrated along the rows and columns. To determine positive pairs, we associate each instance with its patient ID. By design, diagonal elements (green) correspond to the same patient. For illustration purposes, instances 1 and 50 (yellow) belong to the same patient and thus their similarity is also considered in the objective function (see Eq. 3). The blue area corresponds to negative examples as they pertain to instances from different patients.

segments and different projections as positive pairs. For example, in the presence of two temporal segments with timestamps, $t1$ and $t2$, that belong to two leads, $L1$ and $L2$, then $x^i_{t1,L1} = T_{t1,L1}(x^i)$ and $x^i_{t2,L2} = T_{t2,L2}(x^i)$. In Fig. 1 (right), we show this pair of transformed ECG recordings.

We now take a moment to describe Fig. 1 in depth. Given a mini-batch of K instances, the application of a pair of transformation operators results in $2K$ transformed instances. These instances are visualized along the left and top edge of Fig. 1. By calculating the pairwise similarity of representations of these instances, a similarity matrix of dimension $K \times K$ is formed. By associating each instance with its patient ID, a positive pair can be identified. By design, this always includes the diagonal elements. If the same patient reappears within the mini-batch, then off-diagonal elements are also considered.

4.3 Patient-specific Noise Contrastive Estimation Loss

The noise contrastive estimation loss encourages representations in the positive pair to be similar to one another and dissimilar to the remaining representations. Given our patient-centric definition of positive pairs, we encourage a pair of representations, h^i_A and h^k_B where $i, k \in P$, from the same patient, P, to be similar to one another. We quantify this similarity using the cosine similarity, s, with a temperature scaling parameter, τ, as is performed in (Tian et al., 2019; Chen et al., 2020). Such an approach results in the learning of patient-specific representations (see Appendix G).

As shown in Fig. 1, positive pairs always include the diagonal elements. If a patient happens to reappear within the mini-batch, then an off-diagonal positive pair also exists. However, the presence and frequency of these pairs is inconsistent due to the random shuffling of data. Mathematically, our objective function for all pairwise combinations of transformation operators, T_A and T_B, is as follows:

$$\mathcal{L} = \mathbb{E}_{T_A,T_B}\left[\mathcal{L}^{h_A,h_B}_{diagonal} + \mathcal{L}^{h_B,h_A}_{diagonal} + \mathcal{L}^{h_A,h_B}_{off-diagonal} + \mathcal{L}^{h_B,h_A}_{off-diagonal}\right] \tag{1}$$

$$\mathcal{L}^{h_A,h_B}_{diagonal} = \mathbb{E}_{i\in P}\left[\frac{e^{s(h^i_A,h^i_B)}}{\sum_j e^{s(h^i_A,h^j_B)}}\right] \tag{2}$$

$$\mathcal{L}^{h_A,h_B}_{off-diagonal} = \mathbb{E}_{i,k\in P}\left[\frac{e^{s(h^i_A,h^k_B)}}{\sum_j e^{s(h^i_A,h^j_B)}}\right] \tag{3}$$

$$s(h^i_A, h^i_B) = \frac{f_\theta(x^i_A)\cdot f_\theta(x^i_B)}{\|f_\theta(x^i_A)\|\|f_\theta(x^i_B)\|}\frac{1}{\tau} \tag{4}$$

5 Experimental Design

5.1 Datasets

We conduct our experiments using PyTorch (Paszke et al., 2019) on four ECG datasets that include cardiac arrhythmia labels. **PhysioNet 2020** (Perez Alday et al., 2020) consists of 12-lead ECG recordings from 6,877 patients alongside 9 different classes of cardiac arrhythmia. Each recording can be associated with multiple labels. **Chapman** (Zheng et al., 2020) consists of 12-lead ECG recordings from 10,646 patients alongside 11 different classes of cardiac arrhythmia. As is suggested by Zheng et al. (2020), we group these labels into 4 major classes. **PhysioNet 2017** (Clifford et al., 2017) consists of 8,528 single-lead ECG recordings alongside 4 different classes. **Cardiology** (Hannun et al., 2019) consists of single-lead ECG recordings from 328 patients alongside 12 different classes of cardiac arrhythmia. An in-depth description of these datasets can be found in Appendix A.1.

All datasets were split into training, validation, and test sets according to patient ID using a 60, 20, 20 configuration. In other words, patients appeared in only one of the sets. The exact number of instances used during self-supervised pre-training and supervised training can be found in Appendix A.2.

5.2 PRE-TRAINING IMPLEMENTATION

We conduct our pre-training experiments on the training set of two of the four datasets: PhysioNet 2020 and Chapman. We chose these datasets as they contain multi-lead data. The implementation details of the various pre-training methods are as follows. In **CMSC**, we extract a pair of non-overlapping temporal segments of $S = 2500$ samples. Therefore, our model is presented with a mini-batch of dimension $K \times S \times 2$ where K is the batchsize, and S is the number of samples. In **CMLC**, we explore two scenarios with a different number of leads corresponding to the same instance. Our mini-batch dimension is $K \times S \times L$, where L is the number of leads. Lastly, in **CMSMLC**, we incorporate an additional temporal segment in each mini-batch. Therefore, our mini-batch dimension is $K \times 2S \times L$. To ensure a fair comparison between all methods, we expose them to an equal number of patients and instances during training. In CMLC or CMSMLC, we either pre-train using 4 leads (II, V2, aVL, aVR) or all 12 leads. We chose these 4 leads as they cover a large range of axes.

5.3 EVALUATION ON DOWNSTREAM TASK

We evaluate our pre-trained methods in two scenarios. In **Linear Evaluation of Representations**, we are interested in evaluating the utility of the fixed feature extractor in learning representations. Therefore, the pre-trained parameters are frozen and multinomial logistic regression is performed on the downstream supervised task. In **Transfer Capabilities of Representations**, we are interested in evaluating the inductive bias introduced by pre-training. Therefore, the pre-trained parameters are used as an initialization for training on the downstream supervised task.

5.4 BASELINES

We compare our pre-training methods to networks that are initialized randomly (**Random Init.**) or via supervised pre-training (**Supervised**). We also compare to **BYOL** (Grill et al., 2020) and **SimCLR** (Chen et al., 2020), which encourage representations of instances and their perturbed counterparts to be similar to one another, with the aim of learning transformation-invariant representations that transfer well. As SimCLR has been shown to be highly dependent on the choice of perturbations, we explore the following time-series perturbations (see Appendix B for visualizations):

- **Gaussian** - we add $\epsilon \sim \mathcal{N}(0, \sigma)$ to the time-series signal where we chose σ based on the amplitude of the signal. This was motivated by the work of Han et al. (2020) who recently showed the effect of additive noise on ECG signals.
- **Flip** - we flip the time-series signal temporally (**Flip**$_Y$), reversing the arrow of time, or we invert the time-series signal along the x-axis (**Flip**$_X$).
- **SpecAugment** (Park et al., 2019) - we take the short-time Fourier transform of the time-series signal, generating a spectrogram. We then mask either temporal (**SA**$_t$) or spectral (**SA**$_f$) bins of varying widths before converting the spectrogram to the time domain. We also explore the application of sequential perturbations to the time-series signal.

5.5 HYPERPARAMETERS

During self-supervised pre-training, we chose the temperature parameter, $\tau = 0.1$, as per Chen et al. (2020). For BYOL, we chose the decay rate, $\tau_d = 0.90$, after experimenting with various alternatives (see Appendix F). We use the same network architecture for all experiments. Further implementation details can be found in Appendix C.

6 EXPERIMENTAL RESULTS

6.1 LINEAR EVALUATION OF REPRESENTATIONS

In this section, we evaluate the utility of the self-supervised representations learned using four leads on a downstream linear classification task. In Table 1, we show the test AUC on Chapman and PhysioNet 2020 using 50% of the labelled data ($F = 0.5$) after having learned representations, with dimension $E = 128$, using the same two datasets.

We show that CMSC outperforms BYOL and SimCLR on both datasets. On the Chapman dataset, CMSC and SimCLR achieve an $\text{AUC} = 0.896$ and 0.738, respectively, illustrating a 15.8% improvement. Such a finding implies that the representations learned by CMSC are richer and thus allow for improved generalization. We hypothesize that this is due to the setup of CMSC whereby the shared context is across segments (temporally) and patients. Moreover, we show that CLOCS (all 3 proposed methods) outperforms SimCLR in 100% of all conducted experiments, even when pre-training and evaluating with all 12 leads (see Appendix D).

Table 1: Test AUC of the linear evaluation of the representations at $F = 0.5$, after having pre-trained on Chapman or PhysioNet 2020 with $E = 128$. Pre-training and evaluating multi-lead datasets* using 4 leads (II, V2, aVL, aVR). Mean and standard deviation are shown across 5 seeds.

Dataset	Chapman*	PhysioNet 2020*
BYOL	0.643 ± 0.043	0.595 ± 0.018
SimCLR	0.738 ± 0.034	0.615 ± 0.014
CMSC	$\mathbf{0.896 \pm 0.005}$	$\mathbf{0.715 \pm 0.033}$
CMLC	0.870 ± 0.022	0.596 ± 0.008
CMSMLC	0.847 ± 0.024	0.680 ± 0.008

6.2 Effect of Perturbations on Performance

So far, we have presented CLOCS without having incorporated any perturbations during pre-training. However, contrastive learning methods, and in particular SimCLR, are notorious for their over-dependence on the choice of perturbations. To explore this dependence, we apply a diverse set of stochastic perturbations, P, (see Appendix B) during pre-training and observe its effect on generalization performance. We follow the setup introduced by Chen et al. (2020) and apply either a **single perturbation** to each instance, x^i, whereby $x_1^i = P_1(x^i)$, or **sequential perturbations** whereby $x_{1,2}^i = P_2(P_1(x^i))$.

We apply such perturbations while pre-training with SimCLR or CMSC on PhysioNet 2020 using 4 leads and, in Fig. 2, illustrate the test AUC in the linear evaluation scenario. We show that, regardless of the type and number of perturbations, CMSC continues to outperform SimCLR. For example, the *worst-performing* CMSC implementation (Flip_Y) results in an $\text{AUC} = 0.661$ which is still greater than the *best-performing* SimCLR implementation (Gaussian $\rightarrow$ SA_t) with an $\text{AUC} = 0.636$. In fact, we find that pre-training with CMSC *without* applying any perturbations (see Table 1) still outperforms the best-performing SimCLR implementation. Such a finding suggests that CMSC's already strong performance is more likely to stem from its redefinition of the 'shared context' to include both time and patients than from the choice of perturbations.

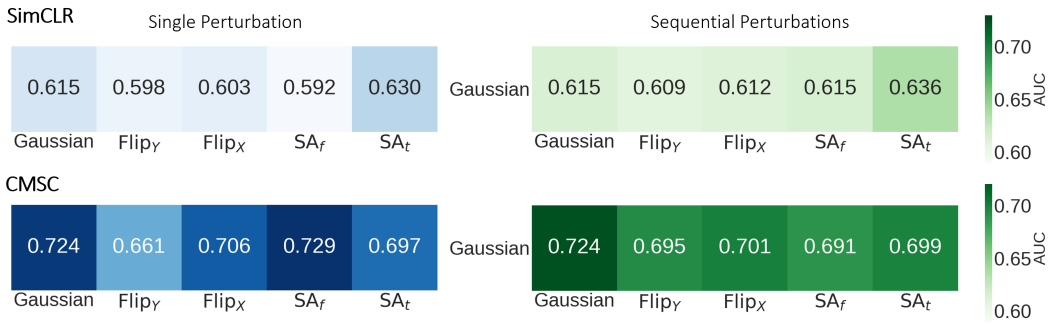


Figure 2: Effect of single (blue) and sequential (green) perturbations applied to the (top) SimCLR and (bottom) CMSC implementations on linear evaluation. Sequential perturbations involve a Gaussian perturbation followed by one of the remaining four types. Pre-training and evaluation was performed on PhysioNet 2020 using 4 leads. Evaluation was performed at $F = 0.5$ and results are averaged across 5 seeds. We show that CMSC outperforms SimCLR regardless of the applied perturbation.

6.3 Transfer Capabilities of Representations

In this section, we evaluate the utility of initializing a network for a downstream task with parameters learned via self-supervision using four leads. In Table 2, we show the test AUC on downstream datasets at $F = 0.5$ for the various self-supervised methods with $E = 128$.

We show that, with a few exceptions, self-supervision is advantageous relative to a Random Initialization. This can be seen by the higher AUC achieved by the former relative to the latter. We also show that, depending on the downstream dataset, either CMSC or CMSMLC outperform BYOL and SimCLR. For example, when pre-training on Chapman and fine-tuning on Cardiology, CMSMLC achieves an $\text{AUC} = 0.717$, a 4.1% improvement compared to SimCLR. This implies that by encouraging representations across space, time, and patients to be similar to one another, networks are nudged into a favourable parameter space. In Appendix E.1, we extend these findings and illustrate that CLOCS outperforms SimCLR in at least 75% of all experiments conducted, on average. When pre-training, fine-tuning, and evaluating using all 12 leads, we show that CMSC outperforms all other methods in at least 90% of all experiments conducted (see Appendix E.2).

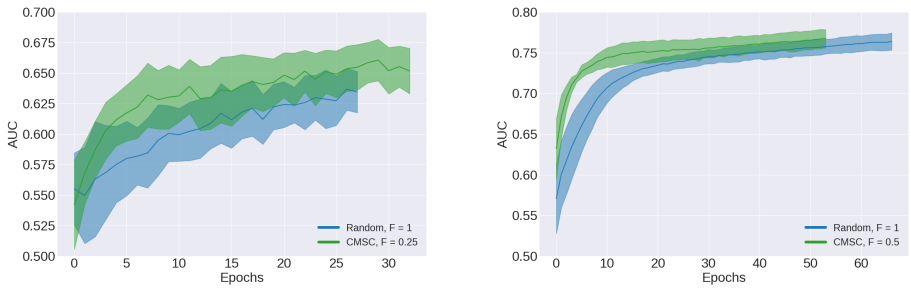
Table 2: Test AUC in the fine-tuning scenario at $F = 0.5$, after having pre-trained on Chapman or PhysioNet 2020 with $E = 128$. Pre-training, fine-tuning, and evaluating multi-lead datasets* using 4 leads. Mean and standard deviation are shown across 5 seeds.

Pretraining Dataset	Chapman*			PhysioNet 2020*		
Downstream Dataset	Cardiology	PhysioNet 2017	PhysioNet 2020*	Cardiology	PhysioNet 2017	Chapman*
Random Init.	0.678 ± 0.011	0.763 ± 0.005	0.803 ± 0.008	0.678 ± 0.011	0.763 ± 0.005	0.907 ± 0.006
Supervised	0.684 ± 0.015	0.799 ± 0.008	0.827 ± 0.001	0.730 ± 0.002	0.810 ± 0.009	0.954 ± 0.003
Self-supervised Pre-training						
BYOL	0.678 ± 0.021	0.748 ± 0.014	0.802 ± 0.013	0.674 ± 0.022	0.757 ± 0.010	0.916 ± 0.009
SimCLR	0.676 ± 0.011	0.772 ± 0.010	0.823 ± 0.011	0.658 ± 0.027	0.762 ± 0.009	0.923 ± 0.010
CMSC	0.695 ± 0.024	0.773 ± 0.013	$\mathbf{0.830 \pm 0.002}$	$\mathbf{0.714 \pm 0.014}$	0.760 ± 0.013	$\mathbf{0.932 \pm 0.008}$
CMLC	0.665 ± 0.016	0.767 ± 0.013	0.810 ± 0.011	0.675 ± 0.013	0.762 ± 0.007	0.910 ± 0.012
CMSMLC	$\mathbf{0.717 \pm 0.006}$	$\mathbf{0.774 \pm 0.004}$	0.814 ± 0.009	0.698 ± 0.011	$\mathbf{0.774 \pm 0.012}$	0.930 ± 0.012

6.4 Doing More With Less Labelled Data

Having established that self-supervision can nudge networks to a favourable parameter space, we set out to investigate whether such a space can lead to strong generalization with less labelled data in the downstream task. In Fig. 3, we illustrate the validation AUC of networks initialized randomly or via CMSC and fine-tuned on two different datasets.

We find that fine-tuning a network based on a CMSC initialization drastically improves data-efficiency. In Fig. 3a, we show that a network initialized with CMSC and exposed to only 25% of the labelled data outperforms one that is initialized randomly and exposed to 100% of the labelled data. This can be seen by the consistently higher AUC during, and at the end of, training. A similar outcome can be seen in Fig. 3b. This suggests that self-supervised pre-training exploits data efficiently such that it can do more with less on downstream classification tasks.

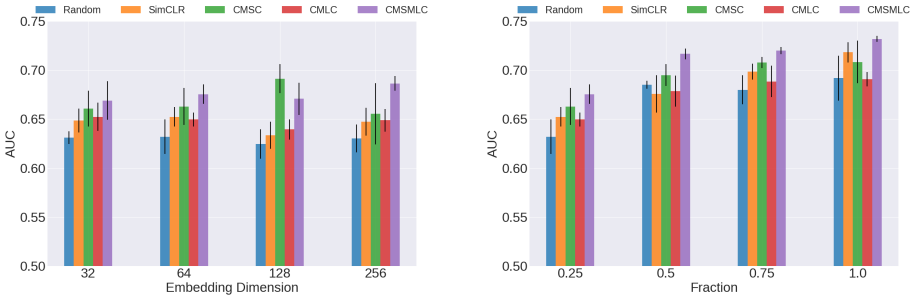


(a) PhysioNet 2020 → Cardiology (b) Chapman → PhysioNet 2017

Figure 3: Validation AUC of a network initialized randomly or via CMSC and which is exposed to different amounts of labelled training data, F. Results are averaged across 5 seeds. Shaded area represents one standard deviation.

6.5 EFFECT OF EMBEDDING DIMENSION, E, AND AVAILABILITY OF LABELLED DATA, F

The dimension of the representation learned during self-supervision and the availability of labelled training data can both have an effect on model performance. In this section, we investigate these claims. In Figs. 4a and 4b, we illustrate the test AUC for all pre-training methods as a function of $E = (32, 64, 128, 256)$ and $F = (0.25, 0.50, 0.75, 1)$.



(a) Chapman → Cardiology, $F = 0.25$ (b) Chapman → Cardiology, $E = 64$

Figure 4: Effect of (a) embedding dimension, E, and (b) labelled fraction, F, on the test AUC when pre-training on Chapman and fine-tuning on Cardiology. Results are averaged across 5 seeds. Error bars represent one standard deviation.

In Fig. 4a, we show that networks initialized randomly or via SimCLR are not significantly affected by the embedding dimension. This can be seen by the $\text{AUC} \approx 0.63$ and ≈ 0.65, for these two methods across all values of E. In contrast, the embedding dimension has a greater effect on CMSC where $\text{AUC} \approx 0.66 \rightarrow 0.69$ as $E = 32 \rightarrow 128$. This implies that CMSC is still capable of achieving strong generalization performance despite the presence of few labelled data ($F = 0.25$).

In Fig. 4b, we show that increasing the amount of labelled training data benefits the generalization performance of all methods. This can be seen by the increasing AUC values as $F = 0.25 \rightarrow 1$. We also show that at all fraction values, CMSMLC outperforms its counterparts. For example, at $F = 1$, CMSMLC achieves an $\text{AUC} = 0.732$ whereas SimCLR achieves an $\text{AUC} = 0.718$. Such superiority still holds at $F = 0.25$ where the two methods achieve an $\text{AUC} = 0.675$ and 0.652, respectively. This outcome emphasizes the robustness of CMSMLC to scarce labelled training data.

7 DISCUSSION AND FUTURE WORK

In this paper, we proposed a family of self-supervised pre-training mechanisms, entitled CLOCS, based on contrastive learning for physiological signals. In the process, we encourage representations across segments (temporally) and leads (spatially) that correspond to instances from the same patient to be similar to one another. We show that our methods outperform the state-of-the-art methods, BYOL and SimCLR, when performing a linear evaluation of and fine-tuning on downstream tasks. This conclusion also holds when applying a range of perturbations and when pre-training and evaluating with a different number of leads. We now elucidate several avenues worth exploring.

Quantifying Patient Similarity. We have managed to learn patient-specific representations. These representations can be used to quantify patient-similarity in order to assist with diagnosis or gain a better understanding of a diseased condition. Validation of these representations can be performed by comparing known similar patients.

Multi-modal Transfer. We transferred parameters from one task to another that shared the same input modality, the ECG. Such data may not always be available for self-supervision. An interesting path would be to explore whether contrastive self-supervision on one modality can transfer well to another modality.

REFERENCES

Philip Bachman, R Devon Hjelm, and William Buchwalter. Learning representations by maximizing mutual information across views. In *Advances in Neural Information Processing Systems*, pp. 15509–15519, 2019.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. *arXiv preprint arXiv:2002.05709*, 2020.

Gari D Clifford, Chengyu Liu, Benjamin Moody, H Lehman Li-wei, Ikaro Silva, Qiao Li, AE Johnson, and Roger G Mark. Af classification from a short single lead ECG recording: the physionet/computing in cardiology challenge 2017. In *2017 Computing in Cardiology*, pp. 1–4, 2017.

Spyros Gidaris, Praveer Singh, and Nikos Komodakis. Unsupervised representation learning by predicting image rotations. *arXiv preprint arXiv:1803.07728*, 2018.

Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre H Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Daniel Guo, Mohammad Gheshlaghi Azar, et al. Bootstrap your own latent: A new approach to self-supervised learning. *arXiv preprint arXiv:2006.07733*, 2020.

Xintian Han, Yuxuan Hu, Luca Foschini, Larry Chinitz, Lior Jankelson, and Rajesh Ranganath. Deep learning models for electrocardiograms are susceptible to adversarial attack. *Nature Medicine*, pp. 1–4, 2020.

Awni Y Hannun, Pranav Rajpurkar, Masoumeh Haghpanahi, Geoffrey H Tison, Codie Bourn, Mintu P Turakhia, and Andrew Y Ng. Cardiologist-level arrhythmia detection and classification in ambulatory electrocardiograms using a deep neural network. *Nature Medicine*, 25(1):65, 2019.

Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. *arXiv preprint arXiv:1911.05722*, 2019.

Aapo Hyvarinen and Hiroshi Morioka. Unsupervised feature extraction by time-contrastive learning and nonlinear ica. In *Advances in Neural Information Processing Systems*, pp. 3765–3773, 2016.

Gustav Larsson, Michael Maire, and Gregory Shakhnarovich. Colorization as a proxy task for visual understanding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6874–6883, 2017.

Yikuan Li, Shishir Rao, Abdelaali Hassaine, Rema Ramakrishnan, Yajie Zhu, Dexter Canoy, Gholamreza Salimi-Khorshidi, Thomas Lukasiewicz, and Kazem Rahimi. Deep bayesian gaussian processes for uncertainty estimation in electronic health records. *arXiv preprint arXiv:2003.10170*, 2020.

Xinrui Lyu, Matthias Hueser, Stephanie L Hyland, George Zerveas, and Gunnar Rätsch. Improving clinical predictions through unsupervised time series representation learning. *arXiv preprint arXiv:1812.00490*, 2018.

Riccardo Miotto, Li Li, Brian A Kidd, and Joel T Dudley. Deep patient: an unsupervised representation to predict the future of patients from the electronic health records. *Scientific Reports*, 6(1): 1–10, 2016.

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

Daniel S Park, William Chan, Yu Zhang, Chung-Cheng Chiu, Barret Zoph, Ekin D Cubuk, and Quoc V Le. Specaugment: A simple data augmentation method for automatic speech recognition. *arXiv preprint arXiv:1904.08779*, 2019.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems*, pp. 8024–8035, 2019.

E. A. Perez Alday, A. Gu, A. Shah, C. Liu, A. Sharma, S. Seyedi, A. Bahrami Rad, M. Reyna, and G. Clifford. Classification of 12-lead ECGs: the PhysioNet - computing in cardiology challenge 2020 (version 1.0.1). *PhysioNet*, 2020.

Pritam Sarkar and Ali Etemad. Self-supervised ecg representation learning for emotion recognition. *arXiv preprint arXiv:2002.03898*, 2020.

Pierre Sermanet, Corey Lynch, Jasmine Hsu, and Sergey Levine. Time-contrastive networks: Self-supervised learning from multi-view observation. In *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 486–487. IEEE, 2017.

Zhiqiang Shen, Zechun Liu, Zhuang Liu, Marios Savvides, and Trevor Darrell. Rethinking image mixture for unsupervised visual representation learning. *arXiv preprint arXiv:2003.05438*, 2020.

Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive multiview coding. *arXiv preprint arXiv:1906.05849*, 2019.

Donglai Wei, Joseph J Lim, Andrew Zisserman, and William T Freeman. Learning and using the arrow of time. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 8052–8060, 2018.

Zhirong Wu, Yuanjun Xiong, Stella X Yu, and Dahua Lin. Unsupervised feature learning via non-parametric instance discrimination. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3733–3742, 2018.

Jianwei Zheng, Jianming Zhang, Sidy Danioko, Hai Yao, Hangyuan Guo, and Cyril Rakovski. A 12-lead electrocardiogram database for arrhythmia research covering more than 10,000 patients. *Scientific Data*, 7(1):1–8, 2020.