Week 4 Theory: Contrastive Self-Supervised Pretraining on MedMNIST

Abstract

This note formalizes the mathematical setting of the self-supervised contrastive pretraining stage ("Week 4") on MedMNIST–style medical image datasets. We define the unlabeled data distribution, the augmentation family, the encoder and projection head, and the SimCLR-style NT-Xent contrastive loss. The goal is to describe the objective that is optimized during self-supervised learning, without referring to any particular implementation or weekly tasks.

1 Unlabeled Data and Augmentations

Let $\mathcal{X} \subset \mathbb{R}^{C\times H\times W}$ denote the space of images (e.g., $C = 1$ for grayscale MedMNIST2D, $H = W = 28$ or 64 after resizing). We assume access to a finite dataset of images

$$\mathcal{D} = \{x_i\}_{i=1}^{N} \subset \mathcal{X}$$

that are originally accompanied by labels (e.g. benign vs. malignant, pneumonia vs. normal), but *labels are not used* during self-supervised pretraining.

Definition 1 (Augmentation family)**.** *Let $\mathcal{T}$ be a family of label-preserving transformations*

$$t : \mathcal{X} \to \mathcal{X},$$

such as random flips, small rotations, random crops, and mild intensity changes, equipped with a probability measure $p(t)$ on $\mathcal{T}$. Sampling $t \sim p$ and applying it to x produces an augmented view $t(x)$.

For any image $x \in \mathcal{X}$, we generate *two* conditionally independent augmented views

$$v^{(1)} = t_1(x), \quad v^{(2)} = t_2(x),$$

where $t_1, t_2 \overset{\text{i.i.d.}}{\sim} p(\cdot)$. These two views are treated as a *positive pair* in the contrastive learning objective.

2 Encoder and Projection Head

The goal of self-supervised pretraining is to learn an encoder that maps images to a representation space in which semantically similar images (in our case, different augmentations of the same underlying image) are close, and dissimilar images are far apart.

Definition 2 (Encoder)**.** *An encoder is a parametric mapping*

$$f_\theta : \mathcal{X} \to \mathbb{R}^d,$$

where θ denotes the learnable parameters (e.g. convolutional filters in a ResNet-18 backbone). For an input $x \in \mathcal{X}$, we denote its encoder feature by

$$h = f_\theta(x) \in \mathbb{R}^d.$$

Following SimCLR-style contrastive learning, we do not apply the contrastive loss directly on h, but instead on the output of a small projection head.

Definition 3 (Projection head)**.** *A projection head is a parametric mapping*

$$g_\phi : \mathbb{R}^d \to \mathbb{R}^k,$$

typically implemented as a small multilayer perceptron (MLP) with parameters ϕ. For a feature vector $h \in \mathbb{R}^d$, its projected representation is

$$z = g_\phi(h) \in \mathbb{R}^k.$$

In the contrastive objective, we use ℓ_2–normalized projected features

$$\tilde{z} = \frac{z}{\|z\|_2} \in \mathbb{R}^k,$$

so that the similarity between two vectors is their cosine similarity.

3 Batch Construction and Similarities

At each optimization step, a mini-batch is constructed by sampling B base images from $\mathcal{D}$:

$$\{x_1, x_2, \ldots, x_B\},$$

and drawing two augmentations for each image:

$$v_i^{(1)} = t_{i,1}(x_i), \quad v_i^{(2)} = t_{i,2}(x_i), \quad i = 1, \ldots, B,$$

where all $t_{i,1}, t_{i,2}$ are i.i.d. according to $p(\cdot)$.

Passing each view through the encoder and projection head yields

$$h_i^{(1)} = f_\theta\big(v_i^{(1)}\big), \qquad z_i^{(1)} = g_\phi\big(h_i^{(1)}\big), \qquad \tilde{z}_i^{(1)} = \frac{z_i^{(1)}}{\|z_i^{(1)}\|_2},$$

$$h_i^{(2)} = f_\theta\big(v_i^{(2)}\big), \qquad z_i^{(2)} = g_\phi\big(h_i^{(2)}\big), \qquad \tilde{z}_i^{(2)} = \frac{z_i^{(2)}}{\|z_i^{(2)}\|_2}.$$

It is convenient to concatenate all $2B$ normalized projected vectors into a single matrix

$$Z = \begin{bmatrix} \tilde{z}_1^{(1)} \\ \vdots \\ \tilde{z}_B^{(1)} \\ \tilde{z}_1^{(2)} \\ \vdots \\ \tilde{z}_B^{(2)} \end{bmatrix} \in \mathbb{R}^{2B \times k}.$$

Definition 4 (Cosine similarity)**.** *For any two normalized embeddings $u, v \in \mathbb{R}^k$ with $\|u\|_2 = \|v\|_2 = 1$, the cosine similarity is*

$$s(u, v) := u^\top v \in [-1, 1].$$

In matrix form, the (p, q)–entry of $S := ZZ^\top \in \mathbb{R}^{2B \times 2B}$ is

$$S_{pq} = s\big(Z_p, Z_q\big) = Z_p^\top Z_q,$$

where Z_p denotes the p-th row of Z.

4 NT-Xent Contrastive Loss

The SimCLR-style NT-Xent loss treats, for each of the $2B$ vectors in the batch, exactly one other vector as its *positive partner*, and all remaining $2B-2$ vectors as *negatives*.

4.1 Indexing of positive pairs

We index the $2B$ views as

$$1, 2, \ldots, B, B+1, \ldots, 2B,$$

with the convention

$$i \longleftrightarrow i+B, \quad \text{for } i = 1, \ldots, B,$$

modulo $2B$, so that each view from the first set is paired with its corresponding view from the second set and vice versa.

Formally, for any index $k \in \{1, \ldots, 2B\}$, define its positive index $p(k)$ by

$$p(k) = \begin{cases} k+B, & \text{if } 1 \le k \le B, \\ k-B, & \text{if } B+1 \le k \le 2B. \end{cases}$$

Then $(k, p(k))$ is a positive pair, whereas any $m \in \{1, \ldots, 2B\} \setminus \{k, p(k)\}$ is treated as a negative for k.

4.2 Per-anchor NT-Xent loss

Let $\tau > 0$ denote the *temperature* parameter, which controls the sharpness of the softmax distribution over similarities. For a fixed anchor index k, the NT-Xent loss is defined as

$$\ell(k) = -\log \frac{\exp\left(\frac{s(Z_k, Z_{p(k)})}{\tau}\right)}{\sum_{\substack{m=1 \\ m \neq k}}^{2B} \exp\left(\frac{s(Z_k, Z_m)}{\tau}\right)}. \tag{1}$$

That is, the numerator corresponds to the similarity between the anchor and its positive partner, and the denominator aggregates exponentiated similarities over all possible candidates (positives and negatives), excluding the anchor itself.

4.3 Batch loss

The NT-Xent loss for the entire mini-batch is the average over all anchors:

$$\mathcal{L}_{\text{NT-Xent}}(\theta, \phi; \mathcal{D}_{\text{batch}}) = \frac{1}{2B} \sum_{k=1}^{2B} \ell(k), \tag{2}$$

where $\mathcal{D}_{\text{batch}}$ denotes the specific images and augmentations used to construct the current mini-batch.

Remark 1 (Symmetry)**.** *The form (2) is symmetric in the sense that each augmented view acts once as anchor and treats its paired view as positive. This is slightly different from using only one direction (e.g. only $k = 1, \ldots, B$), but empirically it helps stabilize training and improves performance.*

4.4 Population objective

At the population level, the self-supervised pretraining objective is defined as the expectation of the batch loss over the data distribution and the augmentation distribution:

$$\mathcal{L}_{\text{SSL}}(\theta, \phi) = \mathbb{E}_{\{x_i\} \sim \mathcal{D}} \mathbb{E}_{\{t_{i,1}, t_{i,2}\} \sim p(\cdot)} \Big[\mathcal{L}_{\text{NT-Xent}}(\theta, \phi; \mathcal{D}_{\text{batch}}) \Big], \tag{3}$$

where the inner expectation is over random augmentations and mini-batch sampling, and the outer expectation is over the empirical dataset.

Self-supervised pretraining aims to find parameters $(\theta^\star, \phi^\star)$ that approximately minimize this objective:

$$(\theta^\star, \phi^\star) \approx \arg\min_{\theta, \phi} \mathcal{L}_{\text{SSL}}(\theta, \phi).$$

5 Geometric Interpretation and Invariance

Intuitively, the NT-Xent objective encourages the following geometric structure in the representation space:

- For each image x, the embeddings of its augmented views, $\tilde{z}^{(1)} = \tilde{z}^{(1)}(x)$ and $\tilde{z}^{(2)} = \tilde{z}^{(2)}(x)$, are encouraged to have high cosine similarity, i.e., to lie close on the unit sphere.
- For two different images x and x', the embeddings of their views are encouraged to have lower similarity, pushing them further apart on the unit sphere.

  Informally, for an ideal encoder f_θ,

$$f_\theta(t(x)) \approx f_\theta(x), \quad \forall x \in \mathcal{X},\ \forall t \in \mathcal{T},$$

up to changes in the projection head g_ϕ and normalization. That is, the representation becomes invariant (or at least robust) to the chosen family of augmentations $\mathcal{T}$.

6 Relation to Mutual Information (Optional View)

The NT-Xent loss can be viewed as a special case of the InfoNCE loss, which provides a lower bound on the mutual information between random variables representing different views of the same underlying instance.

Let $(V^{(1)}, V^{(2)})$ denote the random pair of augmented views of an image, obtained by sampling $x \sim \mathcal{D}$ and $t_1, t_2 \sim p(\cdot)$ independently:

$$V^{(1)} = t_1(x), \quad V^{(2)} = t_2(x).$$

Their embeddings are

$$Z^{(1)} = \tilde{z}^{(1)} = \tilde{z}^{(1)}(x), \quad Z^{(2)} = \tilde{z}^{(2)} = \tilde{z}^{(2)}(x).$$

Under suitable conditions, minimizing the NT-Xent loss corresponds to maximizing a lower bound on the mutual information $I\big(Z^{(1)}; Z^{(2)}\big)$ between the two embeddings, thereby encouraging the encoder to retain information that is *shared* between different augmentations of the same image while discarding augmentation-specific noise.

Although this mutual information view is not strictly necessary for defining the objective, it provides a useful high-level justification for contrastive self-supervision.

7 Summary

In summary, the theory of Week 4 self-supervised pretraining can be expressed as:

- A medical-image dataset $\mathcal{D} = \{x_i\}$ is treated as *unlabeled*.
- A label-preserving augmentation distribution $p(t)$ on $\mathcal{T}$ defines two random views $t_1(x), t_2(x)$ for each x.
- An encoder f_θ and projection head g_ϕ map each view to a normalized embedding on the unit sphere.
- The NT-Xent loss (1)–(2) is computed over batches of $2B$ views, contrasting each embedding with its positive partner and all negatives.
- The self-supervised objective (3) is minimized with respect to (θ, ϕ) via stochastic optimization, yielding an encoder that is approximately invariant to $\mathcal{T}$-augmentations and captures semantically meaningful structure in the data.

These pretrained encoder parameters $\theta^\star$ are then reused in downstream tasks such as linear probing, low-label fine-tuning, and calibration analysis in subsequent weeks.