In a retrieval task, given a triplet (q, d^+, d^-), where:

- q: a query or anchor point,
- d^+: a positive document that is relevant to the query,
- d^-: a negative document that is irrelevant or less relevant to the query,

we can use a triplet margin loss (similarity-based loss) to ensure that d^+ is closer to q than d^- by a specified margin.

The triplet margin loss $\mathcal{L}$ is often formulated as:

$$\mathcal{L}(q, d^+, d^-) = \max(0, \text{sim}(q, d^-) - \text{sim}(q, d^+) + \text{margin})$$

where:

- $\text{sim}(q, d^+)$ is the similarity between q and d^+,
- $\text{sim}(q, d^-)$ is the similarity between q and d^-,
- margin is a positive constant that defines the minimum difference between the similarities.

Explanation of the Loss:

1. **Objective**: The loss penalizes cases where the negative document d^- is closer to q than d^+ by at least the specified margin.
2. **Zero Loss**: If $\text{sim}(q, d^+)$ is already greater than $\text{sim}(q, d^-) + \text{margin}$, the loss is zero (i.e., the model correctly places the positive closer to the query).
3. **Positive Loss**: When $\text{sim}(q, d^+) \leq \text{sim}(q, d^-) + \text{margin}$, the loss increases proportionally to the violation, encouraging the model to adjust representations.

This formulation helps the model learn to rank relevant documents higher than irrelevant ones for a given query in retrieval tasks.

To calculate the gradient of the negative log-likelihood loss $\mathcal{L}_{\text{logistic}}$ with respect to $\mathbf{W}$ based on the similarity function defined as $sim(\mathbf{d}, \mathbf{q}) = \mathbf{q}^T \mathbf{W} \mathbf{d}$, we need to follow these steps:

Step 1: Define the Loss Function

From part (c), the negative log-likelihood loss can be formulated as:

$$\mathcal{L}_{\text{logistic}} = -\log P(\mathbf{d}_p|\mathbf{q}) = -\log \left(\frac{e^{\text{sim}(\mathbf{d}_p, \mathbf{q})}}{e^{\text{sim}(\mathbf{d}_p, \mathbf{q})} + e^{\text{sim}(\mathbf{d}_n, \mathbf{q})}} \right)$$

This simplifies to:

$$\mathcal{L}_{\text{logistic}} = -\text{sim}(\mathbf{d}_p, \mathbf{q}) + \log \left(e^{\text{sim}(\mathbf{d}_p, \mathbf{q})} + e^{\text{sim}(\mathbf{d}_n, \mathbf{q})} \right)$$

Step 2: Substitute the Similarity Function

Substituting $sim(\mathbf{d}, \mathbf{q})$:

$$\mathcal{L}_{\text{logistic}} = -\mathbf{q}^T \mathbf{W} \mathbf{d}_p + \log \left(e^{\mathbf{q}^T \mathbf{W} \mathbf{d}_p} + e^{\mathbf{q}^T \mathbf{W} \mathbf{d}_n} \right)$$

Step 3: Differentiate the Loss Function

To compute the gradient $\nabla_{\mathbf{W}} \mathcal{L}_{\text{logistic}}$, we need to compute the derivatives with respect to $\mathbf{W}$.

1. **Gradient of the first term**:

$$\frac{\partial}{\partial \mathbf{W}} \left(-\mathbf{q}^T \mathbf{W} \mathbf{d}_p \right) = -\mathbf{q} \mathbf{d}_p^T$$

2. **Gradient of the second term**: For the second term, we apply the chain rule. Let $z_p = \mathbf{q}^T \mathbf{W} \mathbf{d}_p$ and $z_n = \mathbf{q}^T \mathbf{W} \mathbf{d}_n$, so we need to differentiate:

$$\log(e^{z_p} + e^{z_n})$$

Using the chain rule:

$$\frac{\partial}{\partial \mathbf{W}} \log(e^{z_p} + e^{z_n}) = \frac{1}{e^{z_p} + e^{z_n}} \cdot \left(e^{z_p} \frac{\partial z_p}{\partial \mathbf{W}} + e^{z_n} \frac{\partial z_n}{\partial \mathbf{W}} \right)$$

- The derivative $\frac{\partial z_p}{\partial \mathbf{W}} = \mathbf{q} \mathbf{d}_p^T$
- The derivative $\frac{\partial z_n}{\partial \mathbf{W}} = \mathbf{q} \mathbf{d}_n^T$

Thus, we can write:

$$\frac{\partial}{\partial \mathbf{W}} \log(e^{z_p} + e^{z_n}) = \frac{1}{e^{z_p} + e^{z_n}} \left(e^{z_p} \mathbf{q} \mathbf{d}_p^T + e^{z_n} \mathbf{q} \mathbf{d}_n^T \right)$$

This simplifies to:

$$\mathbf{q} \left(\frac{e^{z_p} \mathbf{d}_p^T + e^{z_n} \mathbf{d}_n^T}{e^{z_p} + e^{z_n}} \right)$$

The term $\frac{e^{z_p}}{e^{z_p} + e^{z_n}}$ represents the probability $P(\mathbf{d}_p | \mathbf{q})$.

Step 4: Combine the Gradients

Combining the two terms gives us:

$$\nabla_{\mathbf{W}}\mathcal{L}_{\text{logistic}} = -\mathbf{q}\mathbf{d}_p^T + \mathbf{q}P(\mathbf{d}_p|\mathbf{q})\mathbf{d}_p^T + \mathbf{q}P(\mathbf{d}_n|\mathbf{q})\mathbf{d}_n^T$$

Final Gradient Expression

Thus, the final expression for the gradient of the loss with respect to $\mathbf{W}$ is:

$$\nabla_{\mathbf{W}}\mathcal{L}_{\text{logistic}} = \mathbf{q}\left(P(\mathbf{d}_p|\mathbf{q})\mathbf{d}_p^T - \mathbf{d}_p^T + P(\mathbf{d}_n|\mathbf{q})\mathbf{d}_n^T\right)$$

This gradient can now be used in gradient descent algorithms to update $\mathbf{W}$ during the training of the model.

Q2

Part (a)

Any map provided that makes the data linearly separable works. As for a bijective map $f : S^1 \to \mathcal{R}^2$, no such map exists.

Part (b)

1. **True:** The soft margin SVM will converge because the C parameter can be tuned based on the class overlap.
You may also get marks if you write **False** and provide the following reason: Soft margin SVM may not converge or have poor convergence if the C parameter is set too high which essentially makes it a hard margin SVM.

2. Correct options (a) and (c). Following are the justifications:

(a) If C is high, ζ_i's tend to 0, meaning that we are essentially removing the slack introduced in the soft objective converting it to a hard margin objective.

(c) Clearly the solution in fig2(b) seems to be for a hard margin objective or a soft margin objective for a very high value of C. Since in fig2(a) has a solution which mis-classifies a sample, it must be a solution with larger values for slack ζ_i's and correspondingly has smaller value for C.

3. range: $(-1, 1)$ endpoints not included.
Justification: Since the value of $(w^T x + b) = 1$ for a positive support vector and $(w^T x + b) = -1$ for a negative support vector. Any point inside the margin will have a prediction within the given range.