Support Vector Machine – Theoretical Questions

1. Why Does Maximizing Margin Improve Generalization? (Relation to VC Dimension)

For a linear classifier: $f(x) = w^T x + b$

The geometric margin is: $\gamma = \frac{1}{\|w\|}$

Generalization theory shows that the VC dimension of a linear classifier with margin γ in a ball of radius R satisfies:

$$VC \leq \min\left(\frac{R^2}{\gamma^2}, p\right) + 1$$

Thus:

- Larger margin ($\gamma \uparrow$)
- Smaller $\| w \|$
- Lower VC dimension
- Better generalization bounds

Maximizing margin reduces effective capacity and controls overfitting.

2. Derive the Dual Form of SVM Using Lagrange Multipliers

Hard-margin primal problem:

$$\min_{w,b} \frac{1}{2} \| w \|^2$$

subject to:

$$y_i(w^T x_i + b) \geq 1$$

Construct Lagrangian:

$$L(w, b, \alpha) = \frac{1}{2} \parallel w \parallel^2 - \sum_{i=1}^{n} \alpha_i \left[y_i(w^T x_i + b) - 1 \right]$$

where $\alpha_i \geq 0$.

Set derivatives to zero:

$$\frac{\partial L}{\partial w} = 0 \Rightarrow w = \sum \alpha_i y_i x_i$$
$$\frac{\partial L}{\partial b} = 0 \Rightarrow \sum \alpha_i y_i = 0$$

Substitute back:

$$\max_{\alpha} \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j x_i^T x_j$$

subject to:

$$\sum \alpha_i y_i = 0$$
$$\alpha_i \geq 0$$

This is the dual formulation.

3. Why Do Only Support Vectors Matter?

SVM tries to find a line (or hyperplane) that:

- Separates two classes
- Maximizes the margin

The margin is the distance between the boundary and the closest data points.

Support vectors are:

- The data points that lie exactly on the margin
- Or violate the margin (in soft-margin case)

They are the closest points to the boundary.

Why Do They Matter?

Imagine you push the decision boundary slightly.Who will stop it from moving?

Not the far-away points.Only the closest points will restrict the movement.

Those closest points are the support vectors.

From dual solution:

$$w = \sum \alpha_i y_i x_i$$

Only points with $\alpha_i > 0$ contribute.

From KKT (karush–Kuhn–Tucker conditions.) conditions:

$$\alpha_i [y_i (w^T x_i + b) - 1] = 0$$

Thus:

- If point is outside margin → $\alpha_i = 0$
- If point lies on margin → $\alpha_i > 0$

Therefore:

Only support vectors define the hyperplane.

Removing non-support vectors does not change solution.

4. Compare Hinge Loss vs Logistic Loss Mathematically

Step 1: Define Margin

Let: $m = y(w^T x)$

If:

- $m > 0 \rightarrow$ correct classification
- $m < 0 \rightarrow$ wrong classification

Both losses depend on this margin.

Hinge Loss (Used in SVM)

$$\ell_{hinge}(m) = \max(0, 1 - m)$$

Behavior:

- If $m \geq 1 \rightarrow$ loss = 0
- If $m < 1 \rightarrow$ loss increases linearly

It enforces a hard margin rule.

Logistic Loss (Used in Logistic Regression)

$$\ell_{logistic}(m) = \log(1 + e^{-m})$$

Behavior:

- Always positive
- Decreases smoothly as margin increases
- Never exactly zero

It gives probability-based interpretation.

For large margin:

$$\ell_{hinge}(m) = 0$$
$$\ell_{logistic}(m) \approx e^{-m}$$

Hinge explicitly enforces margin ≥ 1. Logistic encourages large margin but never becomes zero.

5. Prove Convexity of the SVM Objective

Soft-margin objective:

$$\min_{w,b} \frac{1}{2} \| w \|^2 + C \sum \max(0, 1 - y_i(w^T x_i + b))$$

Components:

1. $\frac{1}{2} \parallel w \parallel^2$is convex (quadratic).
2. Hinge loss is convex.
3. Sum of convex functions is convex.

Thus SVM objective is convex.

Convex optimization → global minimum exists.

6. How Does Kernel Choice Affect Feature Space Geometry?

Kernel defines inner product:

$$K(x_i, x_j) = \phi(x_i)^T \phi(x_j)$$

Different kernels imply different feature mappings.

Examples:

- Linear kernel → original space.
- Polynomial kernel → polynomial feature expansion.
- RBF kernel → infinite-dimensional feature space.

Thus kernel determines:

- Geometry of separation.
- Shape of decision boundary.
- Complexity of classifier.

RBF creates nonlinear circular boundaries.
Polynomial creates curved surfaces.

7. What Happens When $C \to \infty$?

Soft-margin objective: $\frac{1}{2} \parallel w \parallel^2 + C\sum\xi_i$

If: $C \to \infty$

Misclassification penalty becomes dominant.

Model forces: $\xi_i \to 0$

Thus behaves like hard-margin SVM.

If data not separable → optimization unstable.

Large C → low bias, high variance.

8. Derive SVM from Structural Risk Minimization (SRM)

SRM minimizes: $R_{emp}(f) + \lambda \cdot \Omega(f)$

For SVM:

$$R_{emp} = \sum \text{hinge loss}$$
$$\Omega(f) = \| w \|^2$$

Thus: $\min \frac{1}{2} \| w \|^2 + C\sum \text{hinge loss}$

This balances:

- Empirical error
- Model complexity (margin)

This directly implements Vapnik's Structural Risk Minimization principle.

9. Compare SVM to Neural Networks in High-Dimensional Regimes

SVM:

- Convex optimization
- Unique global solution
- Strong theoretical guarantees
- Effective when p >> n
- Kernel trick allows nonlinear separation

Neural Networks:

- Non-convex optimization
- Multiple local minima
- Large parameter space
- Requires large data
- More flexible function class

In small-data high-dimensional settings:

SVM often more stable.

In large-scale complex problems:

Neural networks outperform.

10. When Does SVM Fail in Practice?

1. Very large datasets (slow training)
2. Poor kernel choice
3. Extreme class imbalance
4. Heavy noise in overlapping regions
5. Requires careful tuning of C and kernel parameters
6. Memory expensive for many support vectors

Also:

If decision boundary highly complex and data massive → deep networks superior.