14. Support Vector Machines 2
STA3142 Statistical Machine Learning

Kibok Lee

Assistant Professor of

Applied Statistics / Statistics and Data Science

Apr 16, 2024

연세대학교
YONSEI UNIVERSITY

Assignment 3

- Due **Friday 5/3, 11:59pm**
- Topics
  - (Programming) K-Nearest Neighbors
  - (Math) MLE vs. MAP
  - (Math) Kernel Methods
  - (Math/Programming) SVM Primal
- **Recommendation: solve math problems before midterm.**
- Please read the instruction carefully!
  - Submit one pdf and one zip file separately
  - Write your code only in the designated spaces
  - Do not import additional libraries
  - …
- If you feel difficult, consider to take **option 2**.

Midterm

- **Tuesday 4/23, 1:10pm — 2:50pm KST**
  - Please come here by 1:00pm!
  - In-person exam
- Closed book with **an A4-size cheat sheet**
  - You can print/write anything on **both side**.
- Coverage: Lec 6—~~13~~**14**
  - True / False, multiple choice, math
- Short practice midterm is available.
  - To be familiar with the type of midterm questions
  - # questions is about a half of the actual exam
  - **No solution will be provided**

Midterm Coverage

- 4,5: Linear Algebra & Probability Review
  - Not main topics, but you should be familiar with them.
  - Some contents (that we feel difficult) can be given FYI.
- 6,7. Linear Regression (and Other Topics)
- 8. Logistic/Softmax Regression
- 9. Generative Classifiers
- 10. Other Classifiers
- 11. Regularization and Validation
- 12. Kernel Methods
- 13, 14. Support Vector Machines
  - From 14, **"how to solve constrained optimization"**

Outline

- Validity of Kernels
- Kernel SVM
  - Constrained Optimization
  - Kernelizing Hard-Margin SVM
  - Kernelizing Soft-Margin SVM
- SVM in Practice

Recap: Kernel Trick

- As we have done, we will embed data $\mathbf{x}$ in a high dimensional space $\phi(\mathbf{x})$, and use simple (linear) models in this space.

- Use algorithms that do not need the coordinates of embeddings $\phi(\mathbf{x})$, but pairwise **inner products**:
$$\phi(\mathbf{x})^T\phi(\mathbf{x}')$$

- Replace these inner products with a **kernel**:
$$k(\mathbf{x},\mathbf{x}') = \phi(\mathbf{x})^T\phi(\mathbf{x}')$$

Recap: Validity of Kernels

1. Prove that there exists a ϕ such that
$$k(\mathbf{x}, \mathbf{x}') = \phi(\mathbf{x})^T \phi(\mathbf{x}'), \forall \mathbf{x}, \mathbf{x}'$$
2. Prove that the Gram matrix K is PSD and use **Mercer's theorem**
   - Note: $\mathrm{PSD} + \mathrm{PSD} = \mathrm{PSD}$ and $c \times \mathrm{PSD} = \mathrm{PSD}$ for $c \geq 0$
   - Also useful to prove if a kernel is invalid; provide a counterexample showing that the Gram matrix K is not PSD
3. Use the axioms provided in previous slides
   - But **not for assignments & exams**; you need to prove them before using them.

Example: Validity of Kernels

- **Q1**. Is $k(\mathbf{x}, \mathbf{z}) = 1 + \mathbf{x}^T\mathbf{z}$ a valid kernel?
  - **Yes.** $\phi(\mathbf{x}) = [1, \mathbf{x}]^T$ then $\phi(\mathbf{x})^T\phi(\mathbf{z}) = k(\mathbf{x}, \mathbf{z})$
  - **Yes.** The Gram matrix $K = \mathbf{1}\mathbf{1}^T + \Phi\Phi^T$ is PSD
    (Prove $\mathbf{1}\mathbf{1}^T$ and $\Phi\Phi^T$ are PSD, then their sum is PSD)

- **Q2**. Is $k(\mathbf{x}, \mathbf{z}) = 1 - \mathbf{x}^T\mathbf{z}$ a valid kernel?
  - **No.** Counterexample: $\mathbf{x}_1 = [1,0]^T, \mathbf{x}_2 = [0,1]^T$
    then, the Gram matrix $K = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$ is not PSD
    (Find $\exists \mathbf{a},\ \mathbf{a}^T K \mathbf{a} < 0$ or any negative eigenvalue of K)
  - **No.** The Gram matrix $K = \mathbf{1}\mathbf{1}^T - \Phi\Phi^T$ is not PSD
    (Find $\exists \mathbf{a},\ \mathbf{a}^T K \mathbf{a} < 0$)

Recap: SVM

- Objective function:

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

  - subject to

$$y^{(n)}\left(\mathbf{w}^T \phi\left(\mathbf{x}^{(n)}\right) + b\right) \geq 1, \forall n = 1, \dots, N$$

- This is a constrained optimization problem.
  - We can solve this using Lagrange multipliers. (convex optimization)

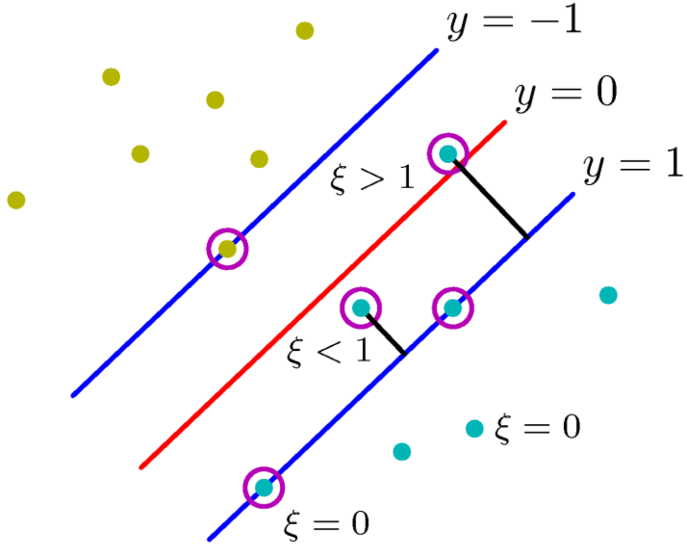
Recap: Soft-Margin SVM

- (Hard-margin) SVM requires an assumption that all data are linearly separable.

$$y^{(n)}\left(\mathbf{w}^T\phi\left(\mathbf{x}^{(n)}\right)+b\right)\geq 1$$

- Soft-margin SVM introduces slack variables $\xi^{(n)}$ for each data point:

$$y^{(n)}\left(\mathbf{w}^T\phi\left(\mathbf{x}^{(n)}\right)+b\right)\geq 1-\xi^{(n)}$$



Recap: Primal Optimization

- Soft-margin SVM:

$$\min_{\mathbf{w},b,\xi} C\sum_{n=1}^{N}\xi^{(n)} + \frac{1}{2}\|\mathbf{w}\|^2$$

  - subject to $y^{(n)}h\left(\mathbf{x}^{(n)}\right) \geq 1-\xi^{(n)}, \forall n$
    $\xi^{(n)} \geq 0, \forall n$

- This is a constrained optimization problem.
  - We can solve this using **Lagrange multipliers**. (convex optimization)
  - Lagrange multipliers convert the constraint into a penalty function.

Recap: Primal Optimization

- Soft-margin SVM:

$$\min_{\mathbf{w},b,\xi} C \sum_{n=1}^{N} \xi^{(n)} + \frac{1}{2} \|\mathbf{w}\|^2$$

  - subject to $y^{(n)} h\left(\mathbf{x}^{(n)}\right) \geq 1 - \xi^{(n)}, \forall n$

$$\xi^{(n)} \geq 0, \forall n$$

- **Lagrangian formulation**:

$$\min_{\mathbf{w},b} C \sum_{n=1}^{N} \max\left(0, 1 - y^{(n)} h\left(\mathbf{x}^{(n)}\right)\right) + \frac{1}{2} \|\mathbf{w}\|^2$$

  - This can be optimized using gradient-based methods!
    - Batch gradient descent (BGD)
    - Stochastic gradient descent (SGD)

Dual Optimization

Dual Optimization

- **Primal** optimization requires a direct access to feature mappings $\phi(\mathbf{x})$.
- We can kernelize SVM to remove explicit $\phi(\mathbf{x})$.
  - This formulation is called **Dual** formulation.
  - In this case, you can use any kernel function, such as polynomial, radial basis function (RBF), and so on.
- With dual variables $a^{(n)}$, we have the followings:

$$\mathbf{w} = \sum_{n=1}^{N} a^{(n)} y^{(n)} \phi\left(\mathbf{x}^{(n)}\right)$$

$$h(\mathbf{x}) = \mathbf{w}^T \phi(\mathbf{x}) + b = \sum_{n=1}^{N} a^{(n)} y^{(n)} k\left(\mathbf{x}, \mathbf{x}^{(n)}\right) + b$$

Kernelizing Hard-Margin SVM

- Objective function:

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

  - subject to

$$y^{(n)}\left(\mathbf{w}^T\phi\left(\mathbf{x}^{(n)}\right)+b\right) \geq 1, \forall n=1,\dots,N$$

- This is a constrained optimization problem.
  - We can solve this using **Lagrange multipliers**. (convex optimization)
  - Kernelization can naturally be done by deriving dual optimization problem.

Constrained Optimization

Constrained Optimization

- General **constrained optimization problem** has the form:

$$\begin{aligned} \min_{\mathbf{x}} \quad & f(\mathbf{x}) \\ \text{subject to} \quad & g_i(\mathbf{x}) \leq 0,\ i = 1, \ldots, m \\ & h_i(\mathbf{x}) = 0,\ i = 1, \ldots, p \end{aligned}$$

- If $\mathbf{x}$ satisfies all the constraints, $\mathbf{x}$ is called **feasible**.

Lagrangian Function

- The **Lagrangian function** of the general constrained optimization is

$$\mathcal{L}(\mathbf{x}, \lambda, \nu) = f(\mathbf{x}) + \sum_{i=1}^{m} \lambda_i g_i(\mathbf{x}) + \sum_{i=1}^{p} \nu_i h_i(\mathbf{x})$$

  - where $\lambda = [\lambda_1, \dots, \lambda_m]$ $(\lambda_i \geq 0, \forall i)$ and $\nu = [\nu_1, \dots, \nu_p]$ are **Langrange multipliers**. (or dual variables)
- Intuitively, Langrange multipliers **penalize** violation of constraints by λ and ν.
- This leads to **primal optimization** problem.

$$\min_{\mathbf{x}} \boxed{\max_{\nu, \lambda : \lambda_i \geq 0, \forall i}} \mathcal{L}(\mathbf{x}, \lambda, \nu)$$

Penalize violation of constraints

Primal and Feasibility

- Primal optimization problem

$$p^* = \min_{\mathbf{x}} \max_{\nu, \lambda : \lambda_i \geq 0, \forall i} \mathcal{L}(\mathbf{x}, \lambda, \nu)$$

  - where $\mathcal{L}(\mathbf{x}, \lambda, \nu) = f(\mathbf{x}) + \sum_{i=1}^{m} \lambda_i g_i(\mathbf{x}) + \sum_{i=1}^{p} \nu_i h_i(\mathbf{x})$

- Note that

$$\mathcal{L}_p(\mathbf{x}) = \max_{\nu, \lambda : \lambda_i \geq 0, \forall i} \mathcal{L}(\mathbf{x}, \lambda, \nu) = \begin{cases} f(\mathbf{x}) & \text{if x is feasible} \\ \infty & \text{otherwise} \end{cases}$$

  - If $\mathbf{x}$ is not feasible, $\exists g_i(\mathbf{x}) > 0$ or $\exists h_i(\mathbf{x}) \neq 0$, such that $\lambda_i \to \infty$ or $\nu_i \to \pm\infty$ leads $\mathcal{L} \to \infty$.

Example: Primal SVM

- Soft-margin SVM:

$$\min_{\mathbf{w},b,\xi} C \sum_{n=1}^{N} \xi^{(n)} + \frac{1}{2}\|\mathbf{w}\|^2$$

  - subject to $y^{(n)}h\left(\mathbf{x}^{(n)}\right) \geq 1 - \xi^{(n)}, \forall n$

$$\xi^{(n)} \geq 0, \forall n$$

- **Lagrangian formulation**:

$$\min_{\mathbf{w},b} C \sum_{n=1}^{N} \max\left(0,1 - y^{(n)}h\left(\mathbf{x}^{(n)}\right)\right) + \frac{1}{2}\|\mathbf{w}\|^2$$

  - Let's check this with the general Langrange multipliers recipe.

Example: Primal SVM

- Soft-margin SVM:

$$\min_{\mathbf{w},b,\boldsymbol{\xi}} C \sum_{n=1}^{N} \xi^{(n)} + \frac{1}{2}\|\mathbf{w}\|^2$$

  - subject to $\max\left(0,1 - y^{(n)} h\left(\mathbf{x}^{(n)}\right)\right) - \xi^{(n)} \leq 0, \forall n$

- Use **Lagrange multipliers** to enforce constraints while optimizing the objective function:

$$\mathcal{L}(\mathbf{w}, b, \boldsymbol{\xi}, \mathbf{a}) = C \sum_{n=1}^{N} \xi^{(n)} + \frac{1}{2}\|\mathbf{w}\|^2 + \sum_{n=1}^{N} a^{(n)} \left\{\max\left(0,1 - y^{(n)} h\left(\mathbf{x}^{(n)}\right)\right) - \xi^{(n)}\right\}$$

  - where $\boldsymbol{a} = \left[a^{(1)}, \dots, a^{(N)}\right] \left(a^{(n)} \geq 0, \forall n\right)$ are **Langrange multipliers**. (or dual variables)

Example: Primal SVM

- Lagrangian function:

$$\mathcal{L}(\mathbf{w}, b, \boldsymbol{\xi}, \mathbf{a}) = C\sum_{n=1}^{N} \xi^{(n)} + \frac{1}{2}\|\mathbf{w}\|^2 + \sum_{n=1}^{N} a^{(n)} \left\{\max\left(0,1 - y^{(n)} h\left(\mathbf{x}^{(n)}\right)\right) - \xi^{(n)}\right\}$$

  - where $\mathbf{a} = \left[a^{(1)}, \dots, a^{(N)}\right] \left(a^{(n)} \geq 0, \forall n\right)$

- Primal optimization problem:

$$\min_{\mathbf{w}, b, \boldsymbol{\xi}} \max_{\mathbf{a}: a^{(n)} \geq 0, \forall n} \mathcal{L}(\mathbf{w}, b, \boldsymbol{\xi}, \mathbf{a})$$

  - First maximize over $\mathbf{a}$, and then minimize over $\mathbf{w}, b, \boldsymbol{\xi}$

- Set the derivative of $\mathcal{L}(\mathbf{w}, b, \boldsymbol{\xi}, \mathbf{a})$ w.r.t $a^{(n)}$ to zero:

$$\frac{\partial \mathcal{L}}{\partial a^{(n)}} = \max\left(0,1 - y^{(n)} h\left(\mathbf{x}^{(n)}\right)\right) - \xi^{(n)} = 0$$

$$\therefore \xi^{(n)} = \max\left(0,1 - y^{(n)} h\left(\mathbf{x}^{(n)}\right)\right)$$

Example: Primal SVM

- Lagrangian function:

$$\mathcal{L}(\mathbf{w}, b, \boldsymbol{\xi}, \mathbf{a}) = C \sum_{n=1}^{N} \xi^{(n)} + \frac{1}{2} \|\mathbf{w}\|^2 + \sum_{n=1}^{N} a^{(n)} \left\{ \max\left(0, 1 - y^{(n)} h\left(\mathbf{x}^{(n)}\right)\right) - \xi^{(n)} \right\}$$

  - where $\mathbf{a} = \left[a^{(1)}, \dots, a^{(N)}\right] \left(a^{(n)} \geq 0, \forall n\right)$

- Substitute $\xi^{(n)} = \max\left(0, 1 - y^{(n)} h\left(\mathbf{x}^{(n)}\right)\right)$:

$$\max_{\mathbf{a}: a^{(n)} \geq 0, \forall n} \mathcal{L}(\mathbf{w}, b, \boldsymbol{\xi}, \mathbf{a}) = C \sum_{n=1}^{N} \max\left(0, 1 - y^{(n)} h\left(\mathbf{x}^{(n)}\right)\right) + \frac{1}{2} \|\mathbf{w}\|^2$$

- We already set $\boldsymbol{\xi}$, and $\mathbf{a}$ values don't matter:

$$\min_{\mathbf{w}, b} C \sum_{n=1}^{N} \max\left(0, 1 - y^{(n)} h\left(\mathbf{x}^{(n)}\right)\right) + \frac{1}{2} \|\mathbf{w}\|^2$$

Lagrange Dual Problem

- Primal optimization problem

$$\min_{\mathbf{x}} \max_{\nu,\lambda:\lambda_i \geq 0,\forall i} \mathcal{L}(\mathbf{x}, \lambda, \nu)$$

- Dual optimization problem

$$\max_{\nu,\lambda:\lambda_i \geq 0,\forall i} \min_{\mathbf{x}} \mathcal{L}(\mathbf{x}, \lambda, \nu)$$

- Another interpretation of dual:

$$\begin{aligned} \max_{\lambda,\nu} \min_{\mathbf{x}} \quad & \mathcal{L}(\mathbf{x}, \lambda, \nu) \\ \text{subject to} \quad & \lambda_i \geq 0,\ \forall i \end{aligned}$$

- Note that the dual solution does not have to be the same with the primal solution.

Weak Duality

- Claim:
$$\begin{aligned} d^* &= \max_{\lambda,\nu:\lambda_i\geq 0} \min_{\mathbf{x}} \mathcal{L}(\mathbf{x},\lambda,\nu) \\ &\leq \min_{\mathbf{x}} \max_{\lambda,\nu:\lambda_i\geq 0} \mathcal{L}(\mathbf{x},\lambda,\nu) \\ &= p^* \end{aligned}$$

- Difference between p^* and d^* is called **duality gap**.

Weak Duality: Proof

Let $\tilde{\mathbf{x}}$ be feasible. Then for any λ, ν with $\lambda_i \geq 0$,

$$\mathcal{L}(\tilde{\mathbf{x}}, \lambda, \nu) = f(\tilde{\mathbf{x}}) + \sum_i \lambda_i g_i(\tilde{\mathbf{x}}) + \sum_i \nu_i h_i(\tilde{\mathbf{x}}) \leq f(\tilde{\mathbf{x}})$$

Thus, $\tilde{\mathcal{L}}(\lambda, \nu) = \min_{\mathbf{x}} \mathcal{L}(\mathbf{x}, \lambda, \nu) \leq \mathcal{L}(\tilde{\mathbf{x}}, \lambda, \nu) \leq f(\tilde{\mathbf{x}})$.
for any λ, ν with $\lambda_i \geq 0$, any feasible $\tilde{\mathbf{x}}$

Then, $d^* = \max_{\lambda,\nu:\lambda_i \geq 0} \tilde{\mathcal{L}}(\lambda, \nu) \leq f(\tilde{\mathbf{x}})$ for any feasible $\tilde{\mathbf{x}}$

Finally, $d^* = \max_{\lambda,\nu:\lambda_i \geq 0} \tilde{\mathcal{L}}(\lambda, \nu) \leq \min_{\tilde{\mathbf{x}}:\text{feasible}} f(\tilde{\mathbf{x}}) = p^*$

Strong Duality

- If $d^* = p^*$, we say **strong duality** holds.
- Conditions for strong duality:
  - It does not hold in general.
  - It holds for convex problems under mild conditions.
  - Conditions that guarantee strong duality in convex problems are called constraint qualification.
- Two well-known sufficient conditions:
  - Slater’s constraint qualification
  - Karush-Kuhn-Tucker (KKT) conditions

Slater's Constraint Qualification

- Strong duality holds for a convex problem

$$\begin{aligned} \min_{\mathbf{x}} \quad & f(\mathbf{x}) \\ \text{subject to} \quad & g_i(\mathbf{x}) \leq 0,\ i = 1, \ldots, m \\ & h_i(\mathbf{x}) = 0,\ i = 1, \ldots, p \end{aligned}$$

  - if there exists strictly feasible **x**, i.e.,

$$\begin{aligned} \exists \mathbf{x}: \quad & g_i(\mathbf{x}) < 0,\ \forall i = 1, \ldots, m \\ & h_i(\mathbf{x}) = 0,\ \forall i = 1, \ldots, p \end{aligned}$$

  - where f, g_i are convex and h_i are affine.
  - Or, this condition becomes trivial if g_i is affine.
- Slater's condition is a **sufficient** condition for **strong duality** to hold for a convex problem.

KKT Conditions

$$\nabla_{\mathbf{x}} f(\mathbf{x}^*) + \sum_{i=1}^{m} \lambda_i^* \nabla_{\mathbf{x}} g_i(\mathbf{x}^*) + \sum_{i=1}^{p} \nu_i^* \nabla_{\mathbf{x}} h_i(\mathbf{x}^*) = 0 \qquad (1)$$

$$h_i(\mathbf{x}^*) = 0,\ i = 1, \ldots, p \qquad (2)$$

$$g_i(\mathbf{x}^*) \leq 0,\ i = 1, \ldots, m \qquad (3)$$

$$\lambda_i^* \geq 0,\ i = 1, \ldots, m \qquad (4)$$

$$\lambda_i^* g_i(\mathbf{x}^*) = 0,\ i = 1, \ldots, m \qquad (5)$$

KKT Conditions

$$\boxed{\nabla_{\mathbf{x}} f(\mathbf{x}^*) + \sum_{i=1}^{m} \lambda_i^* \nabla_{\mathbf{x}} g_i(\mathbf{x}^*) + \sum_{i=1}^{p} \nu_i^* \nabla_{\mathbf{x}} h_i(\mathbf{x}^*) = 0} \quad (1)$$

$$h_i(\mathbf{x}^*) = 0,\ i = 1, \ldots, p \quad (2)$$

Stationarity

$$(\nabla_{\mathbf{x}} \mathcal{L}(\mathbf{x}, \lambda, \nu)|_{\mathbf{x}=\mathbf{x}^*} = 0)$$

$$g_i(\mathbf{x}^*) \leq 0,\ i = 1, \ldots, m \quad (3)$$

$$\lambda_i^* \geq 0,\ i = 1, \ldots, m \quad (4)$$

$$\lambda_i^* g_i(\mathbf{x}^*) = 0,\ i = 1, \ldots, m \quad (5)$$

KKT Conditions

$$\nabla_{\mathbf{x}} f(\mathbf{x}^*) + \sum_{i=1}^{m} \lambda_i^* \nabla_{\mathbf{x}} g_i(\mathbf{x}^*) + \sum_{i=1}^{p} \nu_i^* \nabla_{\mathbf{x}} h_i(\mathbf{x}^*) = 0 \quad (1)$$

$$h_i(\mathbf{x}^*) = 0,\ i = 1, \ldots, p \quad (2)$$

$$g_i(\mathbf{x}^*) \leq 0,\ i = 1, \ldots, m \quad (3)$$

From primal feasibility

$$\lambda_i^* \geq 0,\ i = 1, \ldots, m \quad (4)$$

$$\lambda_i^* g_i(\mathbf{x}^*) = 0,\ i = 1, \ldots, m \quad (5)$$

KKT Conditions

$$\nabla_{\mathbf{x}} f(\mathbf{x}^*) + \sum_{i=1}^{m} \lambda_i^* \nabla_{\mathbf{x}} g_i(\mathbf{x}^*) + \sum_{i=1}^{p} \nu_i^* \nabla_{\mathbf{x}} h_i(\mathbf{x}^*) = 0 \quad (1)$$

$$h_i(\mathbf{x}^*) = 0,\ i = 1, \ldots, p \quad (2)$$

$$g_i(\mathbf{x}^*) \leq 0,\ i = 1, \ldots, m \quad (3)$$

$$\boxed{\lambda_i^* \geq 0,\ i = 1, \ldots, m \qquad \text{From dual feasibility} \quad (4)}$$

$$\lambda_i^* g_i(\mathbf{x}^*) = 0,\ i = 1, \ldots, m \quad (5)$$

KKT Conditions

$$\nabla_{\mathbf{x}} f(\mathbf{x}^*) + \sum_{i=1}^{m} \lambda_i^* \nabla_{\mathbf{x}} g_i(\mathbf{x}^*) + \sum_{i=1}^{p} \nu_i^* \nabla_{\mathbf{x}} h_i(\mathbf{x}^*) = 0 \quad (1)$$

$$h_i(\mathbf{x}^*) = 0,\ i = 1, \ldots, p \quad (2)$$

$$g_i(\mathbf{x}^*) \leq 0,\ i = 1, \ldots, m \quad (3)$$

$$\lambda_i^* \geq 0,\ i = 1, \ldots, m \quad (4)$$

$$\lambda_i^* g_i(\mathbf{x}^*) = 0,\ i = 1, \ldots, m \quad (5)$$

- The last condition is called complementary slackness.

KKT Conditions

- Assume f, g_i, h_i are differentiable.

- If the original problem is **convex** (f, g_i are convex and h_i are affine) and $\mathbf{x}^*, \lambda^*, \nu^*$ satisfy the KKT conditions, then
  - $\mathbf{x}^*$ is primal optimal.
  - (λ^*, ν^*) is dual optimal.
  - Duality gap is zero, i.e., **strong duality** holds.

KKT Conditions: Proof for Sufficiency

- From KKT conditions,

$$\nabla_{\mathbf{x}} f(\mathbf{x}^*) + \sum_{i=1}^{m} \lambda_i^* \nabla_{\mathbf{x}} g_i(\mathbf{x}^*) + \sum_{i=1}^{p} \nu_i^* \nabla_{\mathbf{x}} h_i(\mathbf{x}^*) = 0 \quad (1)$$

$$\left.\begin{aligned} &h_i(\mathbf{x}^*) = 0,\ i = 1, \dots, p \quad (2) \\ &g_i(\mathbf{x}^*) \le 0,\ i = 1, \dots, m \quad (3) \end{aligned}\right\} \mathbf{x}^* \text{ is primal feasible}$$

$$\lambda_i^* \ge 0,\ i = 1, \dots, m \quad \Rightarrow (\lambda^*, \nu^*) \text{ is dual feasible} \quad (4)$$

$$\lambda_i^* g_i(\mathbf{x}^*) = 0,\ i = 1, \dots, m \quad (5)$$

- $\mathcal{L}(\mathbf{x}, \lambda, \nu)$ is a convex differentiable function. Thus, from (1), $\mathbf{x}^*$ is a minimizer of $\mathcal{L}(\mathbf{x}, \lambda, \nu)$.
- Remember, (2) and (5) will be used later.

KKT Conditions: Proof for Sufficiency

- $\mathbf{x}^*$ is primal feasible and a minimizer of $\mathcal{L}(\mathbf{x}, \lambda, \nu)$.
- (λ^*, ν^*) is dual feasible.
- $d_0 \triangleq \widetilde{\mathcal{L}}(\lambda^*, \nu^*) = \min_{\mathbf{x}} \mathcal{L}(\mathbf{x}, \lambda^*, \nu^*)$
  - Let d_0 be a Lagrangian function value.

KKT Conditions: Proof for Sufficiency

- $\mathbf{x}^*$ is primal feasible and a minimizer of $\mathcal{L}(\mathbf{x}, \lambda, \nu)$.
- (λ^*, ν^*) is dual feasible.
- $$\begin{aligned} d_0 \triangleq \widetilde{\mathcal{L}}(\lambda^*, \nu^*) &= \min_{\mathbf{x}} \mathcal{L}(\mathbf{x}, \lambda^*, \nu^*) \\ &= \mathcal{L}(\mathbf{x}^*, \lambda^*, \nu^*) \\ &= f(\mathbf{x}^*) + \sum_{i=1}^{m} \lambda_i^* g_i(\mathbf{x}^*) + \sum_{i=1}^{p} \nu_i^* h_i(\mathbf{x}^*) \end{aligned}$$

KKT Conditions: Proof for Sufficiency

- $\mathbf{x}^*$ is primal feasible and a minimizer of $\mathcal{L}(\mathbf{x}, \lambda, \nu)$.
- (λ^*, ν^*) is dual feasible.
- $d_0 \triangleq \widetilde{\mathcal{L}}(\lambda^*, \nu^*) = \min_{\mathbf{x}} \mathcal{L}(\mathbf{x}, \lambda^*, \nu^*)$

$$= \mathcal{L}(\mathbf{x}^*, \lambda^*, \nu^*)$$

(2) $h_i(\mathbf{x}^*) = 0,\ i = 1, \dots, p$

$$= f(\mathbf{x}^*) + \boxed{\sum_{i=1}^{m} \lambda_i^* g_i(\mathbf{x}^*)} + \boxed{\sum_{i=1}^{p} \nu_i^* h_i(\mathbf{x}^*)}$$

$$= f(\mathbf{x}^*)$$

(5) $\lambda_i^* g_i(\mathbf{x}^*) = 0,\ i = 1, \dots, m$

KKT Conditions: Proof for Sufficiency

- $\mathbf{x}^*$ is primal feasible and a minimizer of $\mathcal{L}(\mathbf{x}, \lambda, \nu)$.
- (λ^*, ν^*) is dual feasible.
- $$\begin{aligned} d_0 \triangleq \widetilde{\mathcal{L}}(\lambda^*, \nu^*) &= \min_{\mathbf{x}} \mathcal{L}(\mathbf{x}, \lambda^*, \nu^*) \\ &= \mathcal{L}(\mathbf{x}^*, \lambda^*, \nu^*) \\ &= f(\mathbf{x}^*) + \sum_{i=1}^{m} \lambda_i^* g_i(\mathbf{x}^*) + \sum_{i=1}^{p} \nu_i^* h_i(\mathbf{x}^*) \\ &= f(\mathbf{x}^*) \end{aligned}$$
- $$d_0 \triangleq \widetilde{\mathcal{L}}(\lambda^*, \nu^*) \le \underbrace{\max_{\lambda,\nu:\lambda_i \ge 0} \widetilde{\mathcal{L}}(\lambda, \nu) \le \min_{\mathbf{x}:\text{ feasible}} f(\mathbf{x})}_{\text{same proof as in weak duality}} \le f(\mathbf{x}^*) = d_0$$
- $$\max_{\lambda,\nu:\lambda_i \ge 0} \widetilde{\mathcal{L}}(\lambda, \nu) = \min_{\mathbf{x}:\text{ feasible}} f(\mathbf{x}) = d_0$$

Q.E.D.

KKT Conditions: Conclusion

- If a constrained optimization
  - is differentiable and
  - has convex objective function and constraint sets,

- The KKT conditions are **necessary and sufficient conditions** for **strong duality** (= zero duality gap).

General Recipe for Dual Optimization

- Given an original optimization

$$
\begin{aligned}
\min_{\mathbf{x}} \quad & f(\mathbf{x}) \\
\text{subject to} \quad & g_i(\mathbf{x}) \leq 0,\ i = 1, \ldots, m \\
& h_i(\mathbf{x}) = 0,\ i = 1, \ldots, p
\end{aligned}
$$

- Solve dual optimization with **Lagrangian function**:

$$
\begin{aligned}
\max_{\lambda,\nu} \min_{\mathbf{x}} \quad & \mathcal{L}(\mathbf{x}, \lambda, \nu) = f(\mathbf{x}) + \sum_{i=1}^{m} \lambda_i g_i(\mathbf{x}) + \sum_{i=1}^{p} \nu_i h_i(\mathbf{x}) \\
\text{subject to} \quad & \lambda_i \geq 0,\ \forall i
\end{aligned}
$$

- Alternatively, solve dual optimization with **Lagrange dual**:

$$
\begin{aligned}
\max_{\lambda,\nu} \quad & \tilde{\mathcal{L}}(\lambda, \nu) \\
\text{subject to} \quad & \lambda_i \geq 0,\ \forall i
\end{aligned}
\qquad \text{where} \quad \tilde{\mathcal{L}}(\lambda, \nu) = \min_{\mathbf{x}} \mathcal{L}(\mathbf{x}, \lambda, \nu)
$$

Recap: KKT Conditions

- Karush-Kuhn-Tucker (KKT) condition:

$$\nabla_{\mathbf{x}} f(\mathbf{x}^*) + \sum_{i=1}^{m} \lambda_i^* \nabla_{\mathbf{x}} g_i(\mathbf{x}^*) + \sum_{i=1}^{p} \nu_i^* \nabla_{\mathbf{x}} h_i(\mathbf{x}^*) = 0$$

$$h_i(\mathbf{x}^*) = 0,\ i = 1, \dots, p$$

$$g_i(\mathbf{x}^*) \leq 0,\ i = 1, \dots, m$$

$$\lambda_i^* \geq 0,\ i = 1, \dots, m$$

$$\lambda_i^* g_i(\mathbf{x}^*) = 0,\ i = 1, \dots, m$$

- The last condition is called complementary slackness and guarantees the strong duality for convex optimization.

Constrained Optimization for SVM

Kernelizing Hard-Margin SVM

- Objective function:

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

  - subject to

$$y^{(n)}\left(\mathbf{w}^T \phi\left(\mathbf{x}^{(n)}\right) + b\right) \geq 1, \forall n = 1, \dots, N$$

- This is a constrained optimization problem.
  - We can solve this using **Lagrange multipliers**. (convex optimization)
  - Kernelization can naturally be done by deriving dual optimization problem.

Kernelizing Hard-Margin SVM

- Use the **Lagrange multipliers** to enforce constraints while optimizing the objective function:

$$\mathcal{L}(\mathbf{w}, b, \mathbf{a}) = \frac{1}{2}\|\mathbf{w}\|^2 + \sum_{n=1}^{N} a^{(n)}\left\{1 - y^{(n)}\left(\mathbf{w}^T\phi\left(\mathbf{x}^{(n)}\right) + b\right)\right\}$$

- Here, $a^{(n)} \geq 0$ are the **Lagrange multipliers** (or dual variables) for each constraint

$$1 - y^{(n)}\left(\mathbf{w}^T\phi\left(\mathbf{x}^{(n)}\right) + b\right) \leq 0, \forall n = 1, \dots, N$$

Lagrangian and Lagrange Dual

- Lagrangian dual optimization problem:

$$\max_{\mathbf{a}} \min_{\mathbf{w},b} \mathcal{L}(\mathbf{w}, b, \mathbf{a})$$

  - subject to $a^{(n)} \geq 0, \forall n = 1, \dots, N$
  - where $\mathcal{L}(\mathbf{w}, b, \mathbf{a}) = \frac{1}{2}\|\mathbf{w}\|^2 + \sum_{n=1}^{N} a^{(n)}\{1 - y^{(n)}(\mathbf{w}^T \phi(\mathbf{x}^{(n)}) + b)\}$

- We first minimize $\mathcal{L}(\mathbf{w}, b, \mathbf{a})$ with respect to $\mathbf{w}, b$ to get the **Lagrange dual**:

$$\max_{\mathbf{a}} \tilde{\mathcal{L}}(\mathbf{a})$$

  - subject to $a^{(n)} \geq 0, \forall n = 1, \dots, N$
  - where $\tilde{\mathcal{L}}(\mathbf{a}) = \min_{\mathbf{w},b} \mathcal{L}(\mathbf{w}, b, \mathbf{a})$

Marginalizing Primal Variables

- Set the derivatives of $\mathcal{L}(\mathbf{w}, b, \mathbf{a})$ w.r.t. $\mathbf{w}, b$ to zero:

$$\frac{\partial \mathcal{L}}{\partial \mathbf{w}} = 0 \Rightarrow \mathbf{w} = \sum_{n=1}^{N} a^{(n)} y^{(n)} \phi\left(\mathbf{x}^{(n)}\right)$$

$$\frac{\partial \mathcal{L}}{\partial b} = 0 \Rightarrow 0 = \sum_{n=1}^{N} a^{(n)} y^{(n)}$$

- Substitute them to eliminate $\mathbf{w}$ and b:

$$\tilde{\mathcal{L}}(\mathbf{a}) = \sum_{n=1}^{N} a^{(n)} - \frac{1}{2} \sum_{n=1}^{N} \sum_{m=1}^{N} a^{(n)} a^{(m)} y^{(n)} y^{(m)} \phi\left(\mathbf{x}^{(n)}\right)^T \phi\left(\mathbf{x}^{(m)}\right)$$

  - subject to $\sum_{n=1}^{N} a^{(n)} y^{(n)} = 0, a^{(n)} \geq 0, \forall n = 1, \ldots, N$

Dual Hard-Margin SVM (with Kernel)

- Define a kernel: $k\left(\mathbf{x}^{(n)}, \mathbf{x}^{(m)}\right) = \phi\left(\mathbf{x}^{(n)}\right)^T \phi\left(\mathbf{x}^{(m)}\right)$
- **Lagrange dual** with kernel:

$$\tilde{\mathcal{L}}(\mathbf{a}) = \sum_{n=1}^{N} a^{(n)} - \frac{1}{2} \sum_{n=1}^{N} \sum_{m=1}^{N} a^{(n)} a^{(m)} y^{(n)} y^{(m)} k\left(\mathbf{x}^{(n)}, \mathbf{x}^{(m)}\right)$$

  - subject to $\sum_{n=1}^{N} a^{(n)} y^{(n)} = 0, a^{(n)} \geq 0, \forall n = 1, \dots, N$
  - This is **quadratic programming**, a kind of **convex optimization**.

- Once we have $\mathbf{a}$, we don't need $\mathbf{w}$ at test time.

$$h(\mathbf{x}) = \mathbf{w}^T \phi(\mathbf{x}) + b = \sum_{n=1}^{N} a^{(n)} y^{(n)} k\left(\mathbf{x}, \mathbf{x}^{(n)}\right) + \boxed{b}$$

What's the value?

Recovering Bias

- For any support vector $\mathbf{x}^{(n)}$,
$$y^{(n)} h\left(\mathbf{x}^{(n)}\right) = 1$$
- Substitute $h(\mathbf{x}) = \sum_{m \in S} a^{(m)} y^{(m)} k\left(\mathbf{x}, \mathbf{x}^{(m)}\right) + b$:
$$y^{(n)} \left(\sum_{m \in S} a^{(m)} y^{(m)} k\left(\mathbf{x}^{(n)}, \mathbf{x}^{(m)}\right) + b \right) = 1$$
  - where S is the index set of support vectors.
- Multiply by $y^{(n)}$ and sum over n:
$$b = \frac{1}{N_S} \left(y^{(n)} - \sum_{m \in S} a^{(m)} y^{(m)} k\left(\mathbf{x}^{(n)}, \mathbf{x}^{(m)}\right) \right)$$
- Why sum over S instead of the entire dataset?

Support Vectors

- KKT conditions:
  - $\nabla_{\mathbf{w}}\mathcal{L}(\mathbf{w}, b, \mathbf{a}) = 0, \nabla_b\mathcal{L}(\mathbf{w}, b, \mathbf{a}) = 0$
  - $1 - y^{(n)}h\left(\mathbf{x}^{(n)}\right) \le 0$
  - $a^{(n)} \ge 0$
  - $a^{(n)}\left(1 - y^{(n)}h\left(\mathbf{x}^{(n)}\right)\right) = 0$
- From the last one, $a^{(n)} = 0$ or $y^{(n)}h\left(\mathbf{x}^{(n)}\right) = 1$
- That is, **only the support vectors matter**.
  - If $a^{(n)} = 0$, we ignore n-th training data.
  - If $y^{(n)}h\left(\mathbf{x}^{(n)}\right) = 1$, n-th training data is a support vector.
  - Thus, we can sum over support vectors only to get $h(\mathbf{x})$.

Kernelizing Soft-Margin SVM

- Soft-margin SVM:

$$\min_{\mathbf{w},b,\boldsymbol{\xi}} C\sum_{n=1}^{N}\xi^{(n)}+\frac{1}{2}\|\mathbf{w}\|^2$$

  - subject to $y^{(n)}h\left(\mathbf{x}^{(n)}\right)\geq 1-\xi^{(n)}, \forall n$

    $\xi^{(n)}\geq 0, \forall n$

- Support vectors satisfy

$$y^{(n)}h\left(\mathbf{x}^{(n)}\right)=1-\xi^{(n)}$$

Lagrangian and Lagrange Dual

- Lagrangian $\mathcal{L}(\mathbf{w}, b, \boldsymbol{\xi}, \mathbf{a}, \boldsymbol{\mu})$

$$= C\sum_{n=1}^{N}\xi^{(n)} + \frac{1}{2}\|\mathbf{w}\|^2 + \sum_{n=1}^{N} a^{(n)}\{1 - y^{(n)}h(\mathbf{x}^{(n)}) - \xi^{(n)}\} + \sum_{n=1}^{N}\mu^{(n)}(-\xi^{(n)})$$

  - where $\xi^{(n)}, a^{(n)}, \mu^{(n)} \geq 0, \forall n$

- We first minimize $\mathcal{L}(\mathbf{w}, b, \boldsymbol{\xi}, \mathbf{a}, \boldsymbol{\mu})$ with respect to $\mathbf{w}, b, \boldsymbol{\xi}$ to get the **Lagrange dual**:

$$\max_{\mathbf{a}, \boldsymbol{\mu}} \tilde{\mathcal{L}}(\mathbf{a}, \boldsymbol{\mu})$$

  - subject to $a^{(n)}, \mu^{(n)} \geq 0, \forall n = 1, \dots, N$
  - where $\tilde{\mathcal{L}}(\mathbf{a}, \boldsymbol{\mu}) = \min_{\mathbf{w}, b, \boldsymbol{\xi}} \mathcal{L}(\mathbf{w}, b, \boldsymbol{\xi}, \mathbf{a}, \boldsymbol{\mu})$

Marginalizing Primal Variables

- Set the derivatives of $\mathcal{L}$ w.r.t. $\mathbf{w}, b, \boldsymbol{\xi}$ to zero:

$$\frac{\partial \mathcal{L}}{\partial \mathbf{w}} = 0 \Rightarrow \mathbf{w} = \sum_{n=1}^{N} a^{(n)} y^{(n)} \phi\left(\mathbf{x}^{(n)}\right)$$

$$\frac{\partial \mathcal{L}}{\partial b} = 0 \Rightarrow 0 = \sum_{n=1}^{N} a^{(n)} y^{(n)}$$

$$\frac{\partial \mathcal{L}}{\partial \boldsymbol{\xi}} = 0 \Rightarrow a^{(n)} = C - \mu^{(n)}$$

- Substitute them to eliminate $\mathbf{w}, b, \boldsymbol{\xi}$:

$$\tilde{\mathcal{L}}(\mathbf{a}) = \sum_{n=1}^{N} a^{(n)} - \frac{1}{2} \sum_{n=1}^{N} \sum_{m=1}^{N} a^{(n)} a^{(m)} y^{(n)} y^{(m)} \phi\left(\mathbf{x}^{(n)}\right)^T \phi\left(\mathbf{x}^{(m)}\right)$$

  - subject to
    $\sum_{n=1}^{N} a^{(n)} y^{(n)} = 0, 0 \leq a^{(n)} \leq C, \forall n = 1, \dots, N$

Dual Soft-Margin SVM (with Kernel)

- Define a kernel: $k\left(\mathbf{x}^{(n)}, \mathbf{x}^{(m)}\right) = \phi\left(\mathbf{x}^{(n)}\right)^T \phi\left(\mathbf{x}^{(m)}\right)$
- **Lagrange dual** with kernel:

$$\tilde{\mathcal{L}}(\mathbf{a}) = \sum_{n=1}^{N} a^{(n)} - \frac{1}{2} \sum_{n=1}^{N} \sum_{m=1}^{N} a^{(n)} a^{(m)} y^{(n)} y^{(m)} k\left(\mathbf{x}^{(n)}, \mathbf{x}^{(m)}\right)$$

  - subject to
    $\sum_{n=1}^{N} a^{(n)} y^{(n)} = 0, 0 \leq a^{(n)} \leq C, \forall n = 1, \ldots, N$
- This is **quadratic programming**, a kind of **convex optimization**.
- **Sequential minimal optimization (SMO)** is an efficient algorithm designed for SVM (out-of-scope)

KKT Conditions

- Lagrangian $\mathcal{L}(\mathbf{w}, b, \boldsymbol{\xi}, \mathbf{a}, \boldsymbol{\mu})$

$$= C\sum_{n=1}^{N}\xi^{(n)} + \frac{1}{2}\|\mathbf{w}\|^2 + \sum_{n=1}^{N}a^{(n)}\{1 - y^{(n)}h(\mathbf{x}^{(n)}) - \xi^{(n)}\} + \sum_{n=1}^{N}\mu^{(n)}(-\xi^{(n)})$$

  - where $\xi^{(n)}, a^{(n)}, \mu^{(n)} \geq 0, \forall n$

- KKT conditions for $\mathcal{L}(\mathbf{w}, b, \boldsymbol{\xi}, \mathbf{a}, \boldsymbol{\mu})$
    - $\nabla_{\mathbf{w}}\mathcal{L} = 0, \nabla_{b}\mathcal{L} = 0, \nabla_{\xi}\mathcal{L} = 0$
    - $1 - y^{(n)}h(\mathbf{x}^{(n)}) - \xi^{(n)} \leq 0$
    - $-\xi^{(n)} \leq 0$

      Both inequality holds, i.e., primal variables are feasible.
    - $a^{(n)} \geq 0$
    - $\mu^{(n)} \geq 0$

      Both inequality holds, i.e., dual variables are feasible.
    - $a^{(n)}\left(1 - y^{(n)}h(\mathbf{x}^{(n)})\right) = 0$
    - $\mu^{(n)}\xi^{(n)} = 0$

      Complementary slackness condition

SVM in Practice

How to Work with SVM

1. Choose the kernel function and slack cost C
   - They are hyperparameters; need validation

2. Solve the optimization problem (many software packages available) – primal or dual

3. Construct the discriminant function from the support vectors

SVM in Practice

- Linear kernel works well for high-dimensional data.
- Choice of (nonlinear) kernels
  - Gaussian (RBF) or polynomial kernel is commonly used.
  - If simple kernels are ineffective, consider more elaborate kernels.
  - Domain experts can give an assistance in formulating appropriate similarity measures.
- Choice of kernel parameters

$$k(\mathbf{x}, \mathbf{z}) = \exp\left(-\frac{\|\mathbf{x} - \mathbf{z}\|^2}{2\sigma^2}\right)$$

  - e.g., for Gaussian kernel, σ is the distance between neighboring points whose labels will likely affect the prediction of the query point.
  - In the absence of reliable criteria, applications rely on the use of a validation set or cross-validation to set such parameters.

SVM with Deep Learning

- Dual/kernel trick is mostly not necessary; deep learning is a learnable nonlinear mapping.

- vs. Softmax regression
  - Softmax regression is more commonly used with deep neural networks. (linear classifier + cross-entropy loss)
  - SVM is often more effective than Softmax regression for transfer learning, i.e., when reusing pre-trained deep learning models for other classification tasks.

Summary

- Kernel Trick
  - Map data points to higher-dimensional space in order to make them linearly separable.
  - Only inner product is used, so we do not need to represent the mapping explicitly.

- SVM is a max-margin classifier
  - Better generalization ability & less overfitting
  - Solved by convex optimization techniques

Additional resources

- Convex optimization textbook
  - https://web.stanford.edu/~boyd/cvxbook/

- Convex optimization course @ Stanford
  - https://web.stanford.edu/class/ee364a/
  - See Chapter 5 for duality

SVM libraries

- LIBSVM
  - https://www.csie.ntu.edu.tw/~cjlin/libsvm/
  - One of the most popular generic SVM solver (supports nonlinear kernels)
- LIBLINEAR
  - https://www.csie.ntu.edu.tw/~cjlin/liblinear/
  - One of the fastest linear SVM solver (linear kernel)
- SVM*light*
  - http://www.cs.cornell.edu/people/tj/svm_light/
  - Structured outputs, various objective measure (e.g., F1, ROC area), Ranking, etc.
- Scikit-learn (sklearn.svm)
  - https://scikit-learn.org/stable/modules/svm.html

Next: Supervised Learning Review