EECS 545: Machine Learning

Lecture 10. Kernel methods: Kernelizing Support Vector Machines

Honglak Lee and Michał Dereziński

02/09/2022

Overview

- Support Vector Machine (SVM)
- Dual optimization
  - General recipe for constrained optimization
  - hard-margin SVM
  - soft-margin SVM

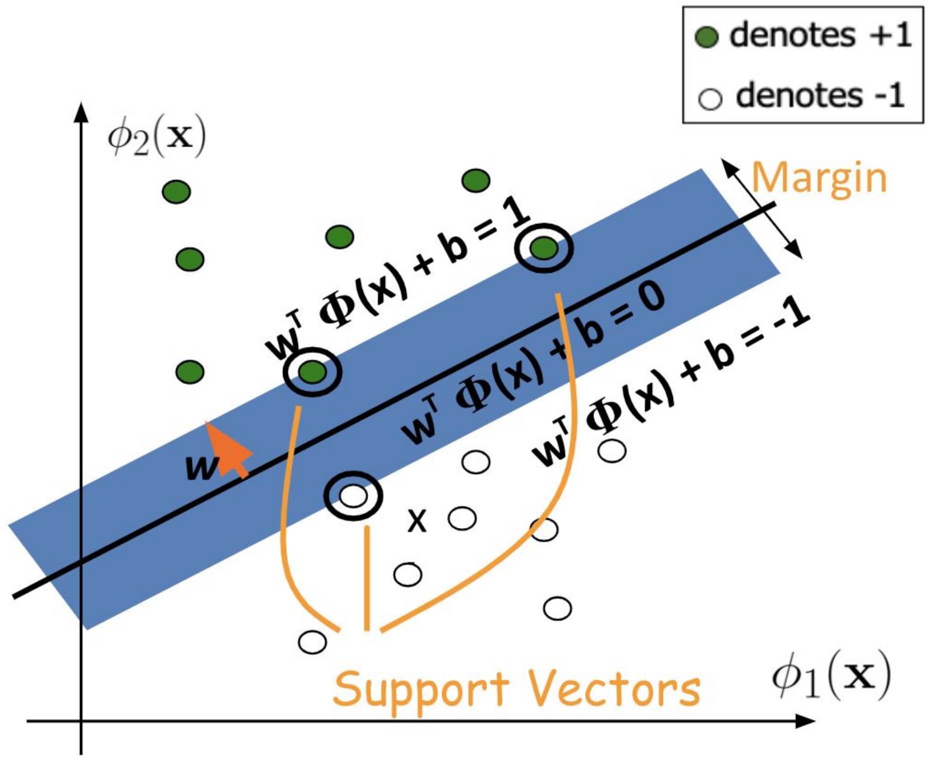
Maximum Margin Classifier

- Optimization problem:

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

subject to

$$\text{For } y^{(n)} = 1, \quad \mathbf{w}^T\phi(\mathbf{x}^{(n)}) + b \geq 1$$
$$\text{For } y^{(n)} = -1, \ \mathbf{w}^T\phi(\mathbf{x}^{(n)}) + b \leq -1$$



Dual optimization

- So far, we have considered primal optimization which requires a direct access to the feature vectors $\phi(\mathbf{x}^{(n)})$
- It is also possible to “kernelize” SVM
  - This formulation is called “Dual” formulation.
  - In this case, you can use any kernel function (such as polynomial, RBF, etc.)

With dual variables $\alpha^{(n)}$,
we have the following relations
(without proofs)

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

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

Kernelizing SVM: back to hard-margin case

- Optimization problem:

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

subject to $$y^{(n)}\left(\mathbf{w}^T\phi\left(\mathbf{x}^{(n)}\right)+b\right)\geq 1, n=1,\cdots,N.$$

- This is a constrained optimization problem.
  - We solve this using Lagrange multipliers (convex optimization)
  - Solving dual optimization problem naturally leads to kernalization

Solving Constrained Optimization: General Overview and Recipe

(This section is just a recap,
see the supplementary
lecture slides for more details)

Constrained Optimization

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

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

- If **x** satisfies all the constraints, **x** is called <u>feasible</u>.
  - In general, this is a nontrivial problem to solve, so we use techniques for convex optimization.

Recap: General Recipe

- Given an original primal optimization

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

add constraint terms with Lagrange multipliers

- Convert to dual problem with Lagrangian function

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

- Solve the dual optimization with Lagrange dual:

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

where $\tilde{\mathcal{L}}(\lambda, \nu) = \min_{\mathbf{x}} \mathcal{L}(\mathbf{x}, \lambda, \nu)$

Recap: A Big Picture

Constrained Optimization Problem

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

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

Lagrangian

e.g. convex objective,
KKT conditions

strong duality (if conditions are met 😎)

$$p^* = d^*$$

Primal Optimization Problem (min-max)

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

weak duality

$$p^* \geq d^*$$

Dual Optimization Problem (max-min)

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

Lagrangian Formulation

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

- The **Lagrangian function** is

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

  – Here, $\lambda = [\lambda_1, \ldots, \lambda_m]$ $(\lambda_i \geq 0, \forall i)$ and $\nu = [\nu_1, \ldots, \nu_p]$ are called Lagrange multipliers (or dual variables)

- This leads to **primal optimization problem**

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

- Difficult to solve directly!

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

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

- Notice that:

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

Lagrange Dual

primal vs dual: switching the order of min / max

Note: these are different problems!

- Dual optimization problem:

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

cf) primal optimization problem

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

- We can also write as:

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

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

Lagrange Dual function

Weak Duality

strong duality (if conditions are met 😎)

$$p^* = d^*$$

Primal Optimization Problem (min-max)

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

weak duality

$$p^* \geq d^*$$

Dual Optimization Problem (max-min)

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

- 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 the **duality gap**.
- In other words, the dual maximization problem (usually easier) gives a "**lower bound**" for the primal minimization problem (usually more difficult).

Weak Duality

Also see Convex Optimization Review Session

- 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,

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

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 $p^* = d^*$, we say strong duality holds.
- What are the conditions for strong duality?
  - does not hold in general
  - holds for convex problems (under mild conditions)
  - conditions that guarantee strong duality in convex problems are called constraint qualification.
- Two well-known conditions
  - Slater's constraint qualification (review session)
  - Karush-Kuhn-Tucker (KKT) condition (main focus)

Conditions for strong duality: Slater's constraint qualification

- Strong duality holds for a convex problem

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

(where *f*, g_i are convex, *and* h_i are affine)

– If it is <u>strictly</u> feasible, i.e.,

$$\exists x : \quad g_i(\mathbf{x}) < 0,\ \forall i = 1, ..., m$$
$$h_i(\mathbf{x}) = 0,\ \forall i = 1, ..., p$$

Slater's condition is a sufficient condition for strong duality to hold for a convex problem

Karush-Kuhn-Tucker (KKT) condition

Let $\mathbf{x}^*$ be a primal optimal and λ^*, ν^* be a dual optimal solution. If the strong duality holds, then we have the following:

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

$$g_i(\mathbf{x}^*) \leq 0, \quad i = 1, \ldots, m,$$ Primal feasibility (2)

$$h_i(\mathbf{x}^*) = 0, \quad i = 1, \ldots, p,$$ Primal feasibility (3)

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

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

$$\begin{array}{ll} \min\limits_{\mathbf{x}} & f(\mathbf{x}) \\ \text{subject to} & g_i(\mathbf{x}) \leq 0,\ i = 1, ..., m \\ & h_i(\mathbf{x}) = 0,\ i = 1, ..., p \end{array}$$

$$\begin{array}{ll} \max\limits_{\lambda,\nu} \min\limits_{\mathbf{x}} & \mathcal{L}(\mathbf{x}, \lambda, \nu) \\ \text{subject to} & \lambda_i \geq 0,\ \forall i \end{array}$$

Dual problem

Note: we do **not** assume the optimization problem is necessarily convex for describing KKT condition. However, when the problem is convex (and differentiable), KKT condition ensures strong duality.

Conditions for strong duality: KKT Conditions

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

- Assume *f*, g_i, h_i are differentiable

- If the original problem is **convex** (where *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, and
  - the <u>duality gap is zero</u> (i.e., strong duality holds)

Proof for sufficiency

Claim: When KKT (1)-(5) holds, the strong duality holds.

- From (2) and (3), $\mathbf{x}^*$ is primal feasible.
- From (4), (λ^*, ν^*) is dual feasible.
- $\mathcal{L}(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\nu})$ is a convex differentiable function. Thus, from (1), $\mathbf{x}^*$ is a minimizer of $\mathcal{L}(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\nu})$.
- Then,

$$
\begin{aligned}
d^* = \tilde{\mathcal{L}}(\lambda^*, \nu^*) &= \min_{\mathbf{x}} \mathcal{L}(\mathbf{x}, \lambda^*, \nu^*) \\
&= \mathcal{L}(\mathbf{x}^*, \lambda^*, \nu^*) \\
&= f(\mathbf{x}^*) + \underbrace{\sum_i \lambda_i^* g_i(\mathbf{x}^*)}_{=0} + \sum_i \nu_i^* h_i(\mathbf{x}^*) \\
&= f(\mathbf{x}^*)
\end{aligned}
$$

(See also: derivation of complementary slackness)

$= 0$ ∵ (5) complementary slackness

- But, $d^* = \tilde{\mathcal{L}}(\lambda^*, \nu^*) \le \underbrace{\max_{\lambda, \nu : \lambda_i \ge 0} \tilde{\mathcal{L}}(\lambda, \nu) \le \min_{\mathbf{x} : \mathbf{x} \text{ is feasible}} f(\mathbf{x})}_{\text{weak duality}} \le f(\mathbf{x}^*) = d^*$

- Then, $$\max_{\lambda, \nu : \lambda_i \ge 0} \tilde{\mathcal{L}}(\lambda, \nu) = \min_{\mathbf{x} : \mathbf{x} \text{ is feasible}} f(\mathbf{x})$$

which proves that the strong duality holds (i.e., duality gap is zero).

KKT conditions: Conclusion

- If a constrained optimization if **differentiable** and has **convex** objective function and constraint sets, then the KKT conditions are **(necessary and) sufficient conditions** for **strong duality** (zero duality gap).

- Thus, the KKT conditions can be used to solve such problems.

Applying Constrained Optimization Techniques for solving SVM

Kernelizing SVM: back to hard-margin case

- Optimization problem:

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

label is either -1 or +1

subject to $$y^{(n)}\left(\mathbf{w}^T\phi\left(\mathbf{x}^{(n)}\right)+b\right) \geq 1, n=1,\cdots,N.$$

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

Back to hard-margin SVM

- Use Lagrange multipliers to enforce constraints while optimizing

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

- Here, $\alpha^{(n)} \geq 0$ is the Lagrange multiplier (or dual variable) for each constraint (one per data point)

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

Lagrangian and Lagrange Dual

- Optimizing the <u>Lagrange</u> dual problem :

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

subject to $\alpha^{(n)} \geq 0, \quad \forall n$

- We first minimize with respect to **w** and b, and get a <u>Lagrange dual problem:</u>

$$\max_{\boldsymbol{\alpha}} \widetilde{L}(\boldsymbol{\alpha})$$

subject to $\alpha^{(n)} \geq 0, \quad \forall n$

(a.k.a. Lagrange dual function)

where $\widetilde{L}(\boldsymbol{\alpha}) = \min_{\mathbf{w},b} L(\mathbf{w}, b, \boldsymbol{\alpha})$

(Please see the supplementary material for more explanation about Lagrange Dual)

Maximize the Margin

- Set the derivatives of $L(\mathbf{w}, b, \boldsymbol{\alpha})$ to zero, to get

$$\mathbf{w} = \sum_{n=1}^{N} \alpha^{(n)} y^{(n)} \phi\left(\mathbf{x}^{(n)}\right) \qquad 0 = \sum_{n=1}^{N} \alpha^{(n)} y^{(n)}$$

c.f. KKT (1) Stationarity

$$\nabla_{\mathbf{w}} L(\mathbf{w}, b, \alpha) = 0$$
$$\nabla_{b} L(\mathbf{w}, b, \alpha) = 0$$

- Substitute in, to eliminate **w** and *b*,

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

subject to $\alpha^{(n)} \geq 0, \quad \forall n$

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

Dual Representation (with kernel)

- Define a kernel $k\left(\mathbf{x}^{(n)}, \mathbf{x}^{(m)}\right) = \phi(\mathbf{x}^{(n)})^T \phi(\mathbf{x}^{(m)})$
- This gives, to maximize

$$\max_{\alpha} \tilde{L}(\alpha) = \sum_{n=1}^{N} \alpha^{(n)} - \frac{1}{2} \sum_{n=1}^{N} \sum_{m=1}^{N} \alpha^{(n)} \alpha^{(m)} y^{(n)} y^{(m)} \underbrace{\phi(\mathbf{x}^{(n)})^T \phi(\mathbf{x}^{(m)})}_{=k(\mathbf{x}^{(n)},\mathbf{x}^{(m)})}$$

subject to $\alpha^{(n)} \geq 0, \quad \forall n$

- Once we have α , we don't need **w**. Predict new values using:

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

Support Vectors

- The KKT conditions are:

$$\nabla_{\mathbf{w}} L(\mathbf{w}, b, \alpha) = 0$$
$$\nabla_{b} L(\mathbf{w}, b, \alpha) = 0$$
$$\alpha^{(n)} \geq 0$$
$$1 - y^{(n)} h\left(\mathbf{x}^{(n)}\right) \leq 0$$
$$\alpha^{(n)} \left\{1 - y^{(n)} h\left(\mathbf{x}^{(n)}\right)\right\} = 0$$

- The last condition (complementary slackness) means:
  - either $\alpha^{(n)} = 0$ or $y^{(n)} h(\mathbf{x}^{(n)}) = 1$.

support vectors

- That is, only the support vectors matter!
  - To compute $h(\mathbf{x})$(prediction), sum only over support vectors

$$h(\mathbf{x}) = \sum_{m:\text{support vectors}} \alpha^{(m)} y^{(m)} k(\mathbf{x}, \mathbf{x}^{(m)}) + b$$

Recovering b

- For any support vector $\mathbf{x}^{(n)} : y^{(n)} h\left(\mathbf{x}^{(n)}\right) = 1$

- Replacing with $h(\mathbf{x}) = \sum_{m \in S} \alpha^{(m)} y^{(m)} k(\mathbf{x}, \mathbf{x}^{(m)}) + b$

$$y^{(n)} \left(\sum_{m \in S} \alpha^{(m)} y^{(m)} k\left(\mathbf{x}^{(n)}, \mathbf{x}^{(m)}\right) + b \right) = 1$$

(index) set of support vectors

- Multiply $y^{(n)}$, and sum over n:

$$b = \frac{1}{N_S} \sum_{n \in S} \left(y^{(n)} - \sum_{m \in S} \alpha^{(m)} y^{(m)} k\left(\mathbf{x}^{(n)}, \mathbf{x}^{(m)}\right) \right)$$

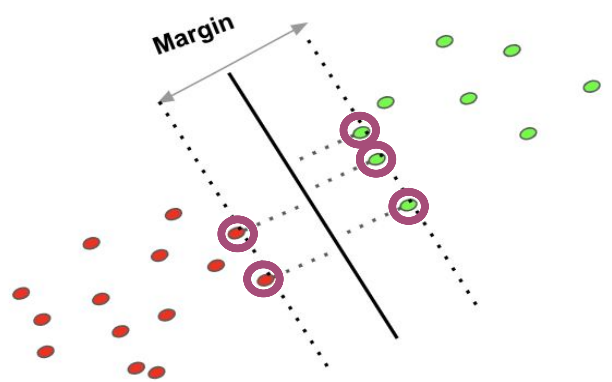


Image adapted from: https://www.vebuso.com/2020/02/a-top-machine-learning-algorithm-explained-support-vector-machines-svms/

Soft SVM

- Maximize the margin, and also penalize for the slack variables

$$C \sum_{n=1}^{N} \xi_n + \frac{1}{2}||\mathbf{w}||^2$$

- The support vectors are now those with

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

Formulation of soft-margin SVM

- Primal optimization
  - Optimization w.r.t. w and $\xi^{(n)}$'s:

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

subject to

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

Dual formulation of soft-margin SVM

- Lagrangian

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

  – where $\alpha^{(n)} \geq 0$, $\mu^{(n)} \geq 0$, $\xi^{(n)} \geq 0, \forall n$

- KKT conditions for the constraints

$$\left.\begin{aligned} 1 - y^{(n)}h\left(\mathbf{x}^{(n)}\right) - \xi^{(n)} &\leq 0 \\ -\xi^{(n)} &\leq 0 \end{aligned}\right\} \text{ Primal variables satisfy the inequality constraints}$$

$$\left.\begin{aligned} \alpha^{(n)} &\geq 0 \\ \mu^{(n)} &\geq 0 \end{aligned}\right\} \text{ Dual variables (for above inequalities) are feasible}$$

$$\left.\begin{aligned} \alpha^{(n)}\left(1 - y^{(n)}h\left(\mathbf{x}^{(n)}\right) - \xi^{(n)}\right) &= 0 \\ \mu^{(n)}\xi^{(n)} &= 0 \end{aligned}\right\} \text{ Complementary slackness condition}$$

Dual formulation of soft-margin SVM

- Taking derivatives

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

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

$$\frac{\partial L}{\partial \xi_n} = 0 \quad \Rightarrow \quad \alpha^{(n)} = C - \mu^{(n)}$$

Dual formulation of soft-margin SVM

$$\mathbf{w} = \sum_{n=1}^{N} \alpha^{(n)} y^{(n)} \phi\left(\mathbf{x}^{(n)}\right) \qquad \sum_{n=1}^{N} \alpha^{(n)} y^{(n)} = 0 \qquad \alpha^{(n)} = C - \mu^{(n)}$$

- Plug these back into the loss:

$$\begin{aligned}
L(\mathbf{w}, b, \xi, \alpha, \mu) &= \tfrac{1}{2}\mathbf{w}^T\mathbf{w} + \textstyle\sum_{n=1}^{N} \underbrace{\left(C - \mu^{(n)}\right)}_{\alpha^{(n)}} \xi^{(n)} + \textstyle\sum_{n=1}^{N} \alpha^{(n)} \{1 - y^{(n)}(\mathbf{w}^T\phi(\mathbf{x}^{(n)}) + b)) - \xi^{(n)}\} \\
&= \tfrac{1}{2}\mathbf{w}^T\mathbf{w} - \textstyle\sum_{n=1}^{N} \alpha^{(n)} y^{(n)} \mathbf{w}^T \phi(\mathbf{x}^{(n)}) - b \underbrace{\sum_{n=1}^{N} \alpha^{(n)} y^{(n)}}_{0} + \textstyle\sum_{n=1}^{N} \alpha^{(n)} \\
&= \frac{1}{2}\mathbf{w}^T\mathbf{w} - \mathbf{w}^T \underbrace{\left(\sum_{n=1}^{N} \alpha^{(n)} y^{(n)} \phi(\mathbf{x}^{(n)})\right)}_{\mathbf{w}} + \sum_{n=1}^{N} \alpha^{(n)} \\
&= \sum_{n=1}^{N} \alpha^{(n)} - \frac{1}{2}\mathbf{w}^T\mathbf{w} \\
&= \sum_{n=1}^{N} \alpha^{(n)} - \frac{1}{2} \sum_{n=1}^{N} \sum_{m=1}^{N} \alpha^{(n)} \alpha^{(m)} y^{(n)} y^{(m)} \phi(\mathbf{x}^{(n)})^T \phi(\mathbf{x}^{(m)})
\end{aligned}$$

Dual formulation of soft-margin SVM

- Dual optimization (via Lagrange dual)

$$\max_{\alpha} \quad \sum_{n=1}^{N} \alpha^{(n)} - \frac{1}{2} \sum_{n=1}^{N} \sum_{m=1}^{N} \alpha^{(n)} \alpha^{(m)} y^{(n)} y^{(m)} k\left(\mathbf{x}^{(n)}, \mathbf{x}^{(m)}\right)$$

Inner product of features replaced with kernel

subject to $\quad 0 \leq \alpha^{(n)} \leq C \qquad \sum_{n=1}^{N} \alpha^{(n)} y^{(n)} = 0$

$$\boxed{\mu^{(n)} = C - \alpha^{(n)} \geq 0}$$

- Solve quadratic problem (convex optimization)

SVM: practical issues

Support Vector Machine: Algorithm

1. Choose a kernel function

2. Choose a value for *C*
   (i.e., smaller C $\longrightarrow$ larger regularization)

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

4. Construct the discriminant function from the support vectors

Some Issues

- Linear kernels work fairly well, but can be suboptimal.

- Choice of (nonlinear) kernels
  – Gaussian or polynomial kernel is default
  – If the simple kernels are ineffective, more elaborate kernels are needed
  – Domain experts can give assistance in formulating appropriate similarity measures

- Choice of kernel parameters
  – E.g., Gaussian kernel: $K(\mathbf{x}, \mathbf{z}) = \exp\left(-\frac{\|\mathbf{x}-\mathbf{z}\|^2}{2\sigma^2}\right)$
    - σ is the distance between neighboring points whose labels will likely to 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.

Summary: Support Vector Machine

- Max margin classifier: improved robustness & less over-fitting
- Solved by convex optimization techniques
- Kernel trick can learn complex decision boundaries

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

Primal Optimization Problem

strong duality

(e.g., KKT condition)

$$p^* = d^*$$

weak duality

$$p^* \geq d^*$$

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

Dual Optimization Problem (max-min)

$$\begin{aligned} \min_{\mathbf{w}, b, \xi} \quad & C \sum_{n=1}^{N} \xi^{(n)} + \tfrac{1}{2} \|\mathbf{w}\|^2 \\ \text{Subject to} \quad & y^{(n)} h\left(\mathbf{x}^{(n)}\right) \geq 1 - \xi^{(n)}, \; \forall n \\ & \xi^{(n)} \geq 0, \forall n \end{aligned}$$

$$\begin{aligned} \max_{\alpha} \quad & \sum_{n=1}^{N} \alpha^{(n)} - \frac{1}{2} \sum_{n=1}^{N} \sum_{m=1}^{N} \alpha^{(n)} \alpha^{(m)} y^{(n)} y^{(m)} k\left(\mathbf{x}^{(n)}, \mathbf{x}^{(m)}\right) \\ \text{subject to} \quad & 0 \leq \alpha^{(n)} \leq C \\ & \sum_{n=1}^{N} \alpha^{(n)} y^{(n)} = 0 \end{aligned}$$

Additional Resource

- Kernel Methods
  - http://www.kernel-machines.org/
- Convex Optimization
  - http://www.stanford.edu/~boyd/cvxbook/
  - http://www.stanford.edu/class/ee364a/
  - see Chapter 5 (and earlier chapters)

SVM Implementation

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

SVM demo code

- http://www.mathworks.com/matlabcentral/fileexchange/28302-svm-demo

- http://www.alivelearn.net/?p=912

Thank you!

[Click here to take the quiz!]

Next class: Neural Networks