Lagrange Multipliers
Kernel Trick

Nicholas Ruozzi
University of Texas at Dallas

Based roughly on the slides of David Sontag
General Optimization

A mathematical detour, we’ll come back to SVMs soon!

$$\min_{x \in \mathbb{R}^n} f_0(x)$$

subject to:

$$f_i(x) \leq 0, \quad i = 1, \ldots, m$$
$$h_i(x) = 0, \quad i = 1, \ldots, p$$
General Optimization

\[
\min_{x \in \mathbb{R}^n} f_0(x)
\]

subject to:

\[
f_i(x) \leq 0, \quad i = 1, \ldots, m
\]

\[
h_i(x) = 0, \quad i = 1, \ldots, p
\]

\(f_0\) is not necessarily convex
General Optimization

subject to:

$$\min_{x \in \mathbb{R}^n} f_0(x)$$

$$f_i(x) \leq 0, \quad i = 1, \ldots, m$$
$$h_i(x) = 0, \quad i = 1, \ldots, p$$

Constraints do not need to be linear
Lagrangian

\[ L(x, \lambda, \nu) = f_0(x) + \sum_{i=1}^{m} \lambda_i f_i(x) + \sum_{i=1}^{p} \nu_i h_i(x) \]

- Incorporate constraints into a new objective function
- \( \lambda \geq 0 \) and \( \nu \) are vectors of Lagrange multipliers
- The Lagrange multipliers can be thought of as soft constraints
Duality

• Construct a dual function by minimizing the Lagrangian over the primal variables

\[ g(\lambda, \nu) = \inf_x L(x, \lambda, \nu) \]

• \( g(\lambda, \nu) = -\infty \) whenever the Lagrangian is not bounded from below for a fixed \( \lambda \) and \( \nu \)
The Primal Problem

\[
\min_{x \in \mathbb{R}^n} f_0(x)
\]

subject to:

\[
f_i(x) \leq 0, \quad i = 1, \ldots, m
\]
\[
h_i(x) = 0, \quad i = 1, \ldots, p
\]

Equivalently,

\[
\inf_{x} \sup_{\lambda \geq 0, \nu} L(x, \lambda, \nu)
\]
The Dual Problem

\[ \sup_{\lambda \geq 0, \nu} g(\lambda, \nu) \]

Equivalently,

\[ \sup_{\lambda \geq 0, \nu} \inf_x L(x, \lambda, \nu) \]

- The dual problem is always concave, even if the primal problem is not convex
Primal vs. Dual

$$\sup_{\lambda \geq 0, \nu} \inf_x L(x, \lambda, \nu) \leq \inf_x \sup_{\lambda \geq 0, \nu} L(x, \lambda, \nu)$$

- Why?
  - $g(\lambda, \nu) \leq L(x, \lambda, \nu)$ for all $x$
  - $L(x', \lambda, \nu) \leq f_0(x')$ for any feasible $x', \lambda \geq 0$

  • $x$ is feasible if it satisfies all of the constraints

- Let $x^*$ be the optimal solution to the primal problem and $\lambda \geq 0$
  $$g(\lambda, \nu) \leq L(x^*, \lambda, \nu) \leq f_0(x^*)$$
• Under certain conditions, the two optimization problems are equivalent

\[ \sup_{\lambda \geq 0, \nu} \inf_x L(x, \lambda, \nu) = \inf_x \sup_{\lambda \geq 0, \nu} L(x, \lambda, \nu) \]

– This is called strong duality

• If the inequality is strict, then we say that there is a duality gap

– Size of gap measured by the difference between the two sides of the inequality
Slater’s Condition

For any optimization problem of the form

$$\min_{x \in \mathbb{R}^n} f_0(x)$$

subject to:

$$f_i(x) \leq 0, \quad i = 1, \ldots, m$$

$$Ax = b$$

where \(f_0, \ldots, f_m\) are convex functions, strong duality holds if there exists an \(x\) such that

$$f_i(x) < 0, \quad i = 1, \ldots, m$$

$$Ax = b$$
Dual SVM

\[
\min_w \frac{1}{2} \|w\|^2
\]

such that

\[
y_i (w^T x^{(i)} + b) \geq 1, \text{ for all } i
\]

- Note that Slater’s condition holds as long as the data is linearly separable
Dual SVM

\[ L(w, b, \lambda) = \frac{1}{2} w^T w + \sum_i \lambda_i (1 - y_i (w^T x^{(i)} + b)) \]

Convex in \( w \), so take derivatives to form the dual

\[ \frac{\partial L}{\partial w_k} = w_k + \sum_i -\lambda_i y_i x_k^{(i)} = 0 \]

\[ \frac{\partial L}{\partial b} = \sum_i -\lambda_i y_i = 0 \]
Dual SVM

\[ L(w, b, \lambda) = \frac{1}{2} w^T w + \sum_i \lambda_i (1 - y_i (w^T x^{(i)} + b)) \]

Convex in \( w \), so take derivatives to form the dual

\[ w = \sum_i \lambda_i y_i x^{(i)} \]

\[ \sum_i \lambda_i y_i = 0 \]
Dual SVM

\[
\max_{\lambda \geq 0} - \frac{1}{2} \sum_i \sum_j \lambda_i \lambda_j y_i y_j x^{(i)^T} x^{(j)} + \sum_i \lambda_i
\]

such that

\[
\sum_i \lambda_i y_i = 0
\]

• By strong duality, solving this problem is equivalent to solving the primal problem

  – Given the optimal \( \lambda \), we can easily construct \( w \) (\( b \) can be found by complementary slackness)
Complementary Slackness

• Suppose that there is zero duality gap

• Let $x^*$ be an optimum of the primal and $(\lambda^*, \nu^*)$ be an optimum of the dual

\[
f_0(x^*) = g(\lambda^*, \nu^*)
\]
\[
= \inf_x \left[ f_0(x) + \sum_{i=1}^{m} \lambda^*_i f_i(x) + \sum_{i=1}^{p} \nu^*_i h_i(x) \right]
\]
\[
\leq f_0(x^*) + \sum_{i=1}^{m} \lambda^*_i f_i(x^*) + \sum_{i=1}^{p} \nu^*_i h_i(x^*)
\]
\[
= f_0(x^*) + \sum_{i=1}^{m} \lambda^*_i f_i(x^*)
\]
\[
\leq f_0(x^*)
\]
Complementary Slackness

- This means that

\[
\sum_{i=1}^{m} \lambda_i^* f_i(x^*) = 0
\]

- As \( \lambda \geq 0 \) and \( f_i(x_i^*) \), this can only happen if \( \lambda_i^* f_i(x_i^*) = 0 \) for all \( i \)

- Put another way,

  - If \( f_i(x^*) < 0 \) (i.e., the constraint is not tight), then \( \lambda_i^* = 0 \)
  
  - If \( \lambda_i^* > 0 \), then \( f_i(x^*) = 0 \)
  
  - ONLY applies when there is no duality gap
Dual SVM

\[
\max_{\lambda \geq 0} -\frac{1}{2} \sum_i \sum_j \lambda_i \lambda_j y_i y_j x^{(i)T} x^{(j)} + \sum_i \lambda_i
\]

such that

\[
\sum_i \lambda_i y_i = 0
\]

- By complementary slackness, \( \lambda_i^* > 0 \) means that \( x^{(i)} \) is a support vector (can then solve for \( b \) using \( w \))
Dual SVM

\[
\max_{\lambda \geq 0} -\frac{1}{2} \sum_i \sum_j \lambda_i \lambda_j y_i y_j x^{(i)T} x^{(j)} + \sum_i \lambda_i
\]

such that

\[
\sum_i \lambda_i y_i = 0
\]

- Takes \( O(n^2) \) time just to evaluate the objective function
  - Active area of research to try to speed this up
The Kernel Trick

\[
\max_{\lambda \geq 0} - \frac{1}{2} \sum_i \sum_j \lambda_i \lambda_j y_i y_j x^{(i)T} x^{(j)} + \sum_i \lambda_i
\]

such that

\[
\sum_i \lambda_i y_i = 0
\]

- The dual formulation only depends on inner products between the data points
  - Same thing is true if we use feature vectors instead
The Kernel Trick

• For some feature vectors, we can compute the inner products quickly, even if the feature vectors are very large.

• This is best illustrated by example:

- Let $\phi(x_1, x_2) = \begin{bmatrix} x_1 x_2 \\ x_2 x_1 \\ x_1^2 \\ x_2^2 \end{bmatrix}$

- $\phi(x_1, x_2) \cdot \phi(z_1, z_2) = x_1^2 z_1^2 + 2x_1 x_2 z_1 z_2 + x_2^2 z_2^2$

  $\quad = (x_1 z_1 + x_2 z_2)^2$

  $\quad = (x \cdot z)^2$

  Reduces to a dot product in the original space.
The Kernel Trick

- The same idea can be applied for the feature vector $\phi$ of all polynomials of degree (exactly) $d$

$$- \phi(x) \cdot \phi(z) = (x \cdot z)^d$$

- More generally, a kernel is a function

$$k(x, z) = \phi(x) \cdot \phi(z)$$

for some feature map $\phi$

- Rewrite the dual objective

$$\max_{\lambda \geq 0, \sum_i \lambda_i y_i = 0} -\frac{1}{2} \sum_i \sum_j \lambda_i \lambda_j y_i y_j k(x^{(i)}, x^{(j)}) + \sum_i \lambda_i$$
Examples of Kernels

• Polynomial kernel of degree exactly $d$
  \[ k(x, z) = (x \cdot z)^d \]

• General polynomial kernel of degree $d$ for some $c$
  \[ k(x, z) = (x \cdot z + c)^d \]

• Gaussian kernel for some $\sigma$
  \[ k(x, z) = \exp \left( \frac{-||x-z||^2}{2\sigma^2} \right) \]
  – The corresponding $\phi$ is infinite dimensional!

• So many more…
Kernels

• Bigger feature space increases the possibility of overfitting
  – Large margin solutions should still generalize reasonably well

• Alternative: add “penalties” to the objective to disincentivize complicated solutions

\[
\min_{w} \frac{1}{2} \|w\|^2 + c \cdot (\# \text{ of misclassifications})
\]

  – Not a quadratic program anymore (in fact, it’s NP-hard)
  – Similar problem to Hamming loss, no notion of how badly the data is misclassified