COMPSCI 689
Lecture 10: Lagrangian Duality and Kernel SVMs

Benjamin M. Marlin

College of Information and Computer Sciences
University of Massachusetts Amherst

Slides by Benjamin M. Marlin (marlin@cs.umass.edu).

The Lagrangian

- The fundamental tool for analyzing constrained optimization problems is the Lagrangian function.
- Given an objective function $f($**x** and a set of equality and inequality constraint functions $c_i(\mathbf{x}) = 0$ for $i \in \mathcal{E}$ and $c_i(\mathbf{x}) \geq 0$ for $i \in \mathcal{I}$, the Lagrangian is the function:

$$\mathcal{L}(\mathbf{x}, \lambda) = f(\mathbf{x}) - \sum_{i \in \mathcal{I}} \lambda_i c_i(\mathbf{x}) - \sum_{i \in \mathcal{E}} \lambda_i c_i(\mathbf{x})$$

- The new variables λ_i are referred to as *Lagrange Multipliers*.

The KKT Conditions

- The KKT conditions specify necessary first-order conditions on the solution $\mathbf{x}^*$ in terms of the Lagrangian $\mathcal{L}(\mathbf{x}, \lambda)$.
- Suppose that $\mathbf{x}_*$ is a solution to a constrained optimization problem with Lagrangian $\mathcal{L}(\mathbf{x}, \lambda)$ and constraints $c_i(\mathbf{x}) = 0$ for $i \in \mathcal{E}$ and $c_i(\mathbf{x}) \geq 0$ for $i \in \mathcal{I}$. Then there existis a λ_* such that:

$$\begin{aligned}
\nabla_{\mathbf{x}} \mathcal{L}(\mathbf{x}_*, \lambda_*) &= 0 \\
c_i(\mathbf{x}_*) &= 0 \text{ for all } i \in \mathcal{E} \\
c_i(\mathbf{x}_*) &\geq 0 \text{ for all } i \in \mathcal{I} \\
\lambda_{i*} &\geq 0 \text{ for all } i \in \mathcal{I} \\
\lambda_{i*} c_i(\mathbf{x}_*) &= 0 \text{ for all } i \in \mathcal{I}
\end{aligned}$$

- For convex problems, if $(\mathbf{x}_*, \lambda)$ satisfy the KKT conditions, then $\mathbf{x}_*$ is the global constrained optimum.

Method of Lagrange Multipliers

- If we only have equality constraints, to find a point that satisfies the KKT conditions, we identify and analyze the stationary points of the Lagrangian by solving the Lagrangian gradient system:

$$\nabla_{\mathbf{x}}\mathcal{L}(\mathbf{x}, \lambda) = 0$$

$$\nabla_{\lambda}\mathcal{L}(\mathbf{x}, \lambda) = 0$$

- We identify all values of $\mathbf{x}$, and λ satisfying the above equations, plug the values of $\mathbf{x}$ into $f(\mathbf{x})$ and determine the minimizer.
- This is called the method of Lagrange multipliers.

Example: Method of Lagrange Multipliers

Consider the constrained problem shown below.

$$\begin{aligned} \mathbf{x}_* = \arg\min_{\mathbf{x}} x_1 + x_2 \\ \text{s.t. } x_1^2 + x_2^2 = 2 \end{aligned}$$

What is the Lagrangian for this optimization problem?

The Lagrangian is $\mathcal{L}(x_1, x_2, \lambda) = x_1 + x_2 - \lambda(x_1^2 + x_2^2 - 2)$.

Example: Method of Lagrange Multipliers

Given the Lagrangian $\mathcal{L}(x_1, x_2, \lambda) = x_1 + x_2 - \lambda(x_1^2 + x_2^2 - 2)$, to find the optimizer, we solve the Lagrangian gradient system:

$$\frac{\partial \mathcal{L}(x_1, x_2, \lambda)}{\partial x_1} = 1 - 2\lambda x_1 = 0 \Rightarrow x_1 = \frac{1}{2\lambda}$$

$$\frac{\partial \mathcal{L}(x_1, x_2, \lambda)}{\partial x_2} = 1 - 2\lambda x_2 = 0 \Rightarrow x_2 = \frac{1}{2\lambda}$$

$$\frac{\partial \mathcal{L}(x_1, x_2, \lambda)}{\partial \lambda} = x_1^2 + x_2^2 - 2 = 0 \Rightarrow x_1^2 + x_2^2 = 2$$

Plugging the first two results into the third result we find that $\lambda^2 = \sqrt{1/4}$ and thus $\lambda = \pm 1/2$. Plugging this back into the first two results we get that $x_1 = x_2 = \pm 1$. Checking both solutions by plugging into $x_1 + x_2$, the minimizer is $x_1 = x_2 = -1$.

What about inequality constraints?

- If we have a problem with both inequality and equality constraints, we have no method to solve it (yet).
- However, we can ignore the inequality constraints and use the method of Lagrange multipliers with the equality constraints.
- If a set of solutions to the equality constrained problem also satisfies the inequality constraints, then the element of that set with the smallest value of $f(\mathbf{x})$ is the global minimizer.
- If none of the solutions to the equality constrained problem also satisfy the inequality constraints, then this approach does not yield a solution.

What About Inequality Constraints?

- Even linear inequality constraints can make convex optimization problems harder to solve.

- There exist specialized algorithms for solving many different kinds of convex optimization problems including linear programs (LPs), quadratic programs (QPs), second-order cone programs (SOCPs), etc. that include constraints.

- A general strategy in optimization is to transform a problem for which there are no convenient solution methods into a form where the problem can be more easily solved or that exposes additional properties (see SVC margin to hinge loss transformation).

- One often useful transformation of a convex optimization problem with inequality constraints is the *Lagrange Dual*.

The Lagrange Dual

- Suppose we have a constrained optimization problem with objective function $f(x)$ and constraint functions $c_i(\mathbf{x}) \geq 0$ and $c_i(\mathbf{x}) = 0$. We can form the Lagrangian as shown below:

$$\mathcal{L}(\mathbf{x}, \lambda) = f(\mathbf{x}) - \sum_{i \in \mathcal{E}} \lambda_i c_i(\mathbf{x}) - \sum_{i \in \mathcal{I}} \lambda_i c_i(\mathbf{x})$$

- The Lagrange dual function of the original optimization problem is:

$$q(\lambda) = \min_{\mathbf{x} \in \mathcal{X}} \mathcal{L}(\mathbf{x}, \lambda)$$

- This function is subject to the constraints $\lambda_i \geq 0$ for $i \in \mathcal{I}$.

Properties of the Lagrange Dual

- For any value of λ satisfying the constraints, $q(\lambda)$ provides a lower bound on the primal objective function value $f(\mathbf{x}_*)$ at the constrained minimizer $\mathbf{x}_*$. In other words: $f(\mathbf{x}_*) \geq q(\lambda)$.
- If there exists a λ_* such that $f(\mathbf{x}_*) = q(\lambda_*)$, then *strong duality* is said to hold. Otherwise, *weak duality* holds.
- Strong duality typically holds for convex optimization problems with convex constraints.

Properties of Strong Duality

- When strong duality holds, **minimizing** the primal objective function $f(\mathbf{x})$ with respect to $\mathbf{x}$ subject to the primal constraints is exactly equivalent to **maximizing** the Lagrange dual function $q(\lambda)$ with respect to λ subject to the dual constraints [1]:

$$\begin{aligned} \lambda_* = \arg\max_{\lambda} q(\lambda) \\ \text{s.t.} \quad \lambda_i \geq 0 \text{ ... for } i \in \mathcal{I} \end{aligned}$$

- Finally, it is also possible to recover the optimal value of $\mathbf{x}_*$ from λ_*, typically using relationships between them derived in the process of forming the Lagrange dual from the primal.

[1] Note that additional equality constraints may sometimes be produced in the process of forming the dual, and these also need to be included in the dual problem formulation

The SVC Primal Optimization Problem

- Recall that the constrained version of the SVC optimization problem has the form shown below.

$$\begin{aligned} &\arg\min_{\mathbf{w},b,\epsilon} \frac{1}{2}\|\mathbf{w}\|_2^2 + C\sum_{n=1}^{N}\epsilon_n \\ &\text{s.t. } \forall n \;\; y_n(\mathbf{w}\mathbf{x}_n^T + b) \geq 1 - \epsilon_n \\ &\forall n \;\; \epsilon_n \geq 0. \end{aligned}$$

- This problem has a convex objective with convex constraints. We will derive its Lagrange dual.

Lagrangian

$$
\begin{aligned}
&\arg\min_{\mathbf{w},b,\epsilon} \frac{1}{2}\|\mathbf{w}\|_2^2 + C\sum_{n=1}^{N}\epsilon_n \\
&\text{s.t. } \forall n \;\; y_n(\mathbf{w}\mathbf{x}_n^T + b) \geq 1 - \epsilon_n \\
&\quad \forall n \;\; \epsilon_n \geq 0.
\end{aligned}
$$

$$
\begin{aligned}
\mathcal{L}(\mathbf{w}, b, \epsilon, \alpha, \beta) =& \frac{1}{2}\|\mathbf{w}\|_2^2 + C\sum_{n=1}^{N}\epsilon_n \; - \sum_{n=1}^{N}\beta_n\epsilon_n \\
& - \sum_{n=1}^{N}\alpha_n(y_n(\mathbf{w}\mathbf{x}_n^T + b) + \epsilon_n - 1) \\
& \text{s.t. } \forall n \;\; \alpha_n \geq 0, \beta_n \geq 0
\end{aligned}
$$

Minimizing w

$$
\begin{aligned}
\mathcal{L}(\mathbf{w}, b, \epsilon, \alpha, \beta) = & \frac{1}{2}\|\mathbf{w}\|_2^2 + C\sum_{n=1}^{N}\epsilon_n \; - \sum_{n=1}^{N}\beta_n\epsilon_n \\
& - \sum_{n=1}^{N}\alpha_n(y_n(\mathbf{w}\mathbf{x}_n^T + b) + \epsilon_n - 1) \\
& \text{s.t. } \forall n \;\; \alpha_n \geq 0, \beta_n \geq 0
\end{aligned}
$$

$$
\begin{aligned}
\nabla_{\mathbf{w}}\mathcal{L}(\mathbf{w}, b, \epsilon, \alpha, \beta) = \mathbf{w} - \sum_{n=1}^{N}\alpha_n y_n \mathbf{x}_n = 0 \\
\mathbf{w} = \sum_{n=1}^{N}\alpha_n y_n \mathbf{x}_n
\end{aligned}
$$

Minimizing b

$$\mathcal{L}(\mathbf{w}, b, \epsilon, \alpha, \beta) = \frac{1}{2}\|\mathbf{w}\|_2^2 + C\sum_{n=1}^{N}\epsilon_n \; - \sum_{n=1}^{N}\beta_n\epsilon_n$$

$$- \sum_{n=1}^{N}\alpha_n(y_n(\mathbf{w}\mathbf{x}_n^T + b) + \epsilon_n - 1)$$

$$\text{s.t. } \forall n \;\; \alpha_n \geq 0, \beta_n \geq 0$$

$$\nabla_b\mathcal{L}(\mathbf{w}, b, \epsilon, \alpha, \beta) = \; - \sum_{n=1}^{N}\alpha_n y_n = 0$$

$$\sum_{n=1}^{N}\alpha_n y_n = 0$$

Minimizing ϵ

$$\begin{aligned}\mathcal{L}(\mathbf{w}, b, \epsilon, \alpha, \beta) =& \frac{1}{2}\|\mathbf{w}\|_2^2 + C\sum_{n=1}^{N}\epsilon_n \; - \sum_{n=1}^{N}\beta_n\epsilon_n \\ & - \sum_{n=1}^{N}\alpha_n(y_n(\mathbf{w}\mathbf{x}_n^T + b) + \epsilon_n - 1) \\ & \text{s.t. } \forall n \;\; \alpha_n \geq 0, \beta_n \geq 0\end{aligned}$$

$$\begin{aligned}\nabla_{\epsilon_n}\mathcal{L}(\mathbf{w}, b, \epsilon, \alpha, \beta) &= C - \beta_n - \alpha_n = 0 \\ \alpha_n + \beta_n &= C\end{aligned}$$

Substituting into Lagrangian

$$\begin{aligned}\mathcal{L}(\mathbf{w}, b, \epsilon, \alpha, \beta) =& \frac{1}{2}\|\mathbf{w}\|_2^2 + C\sum_{n=1}^{N}\epsilon_n \; - \sum_{n=1}^{N}\beta_n\epsilon_n \\ & - \sum_{n=1}^{N}\alpha_n(y_n(\mathbf{w}\mathbf{x}_n^T + b) + \epsilon_n - 1)\end{aligned}$$

$$\begin{aligned}\mathcal{L}(\mathbf{w}, b, \epsilon, \alpha, \beta) =& \frac{1}{2}\|\sum_{m=1}^{N}\alpha_m y_m\mathbf{x}_m\|_2^2 + C\sum_{n=1}^{N}\epsilon_n \; - \sum_{n=1}^{N}\beta_n\epsilon_n \\ & - \sum_{n=1}^{N}\alpha_n(y_n((\sum_{m=1}^{N}\alpha_m y_m\mathbf{x}_m)\mathbf{x}_n^T + b) + \epsilon_n - 1)\end{aligned}$$

Expanding Terms

$$\begin{aligned}\mathcal{L}(\mathbf{w}, b, \epsilon, \alpha, \beta) =& \frac{1}{2} \| \sum_{m=1}^{N} \alpha_m y_m \mathbf{x}_m \|_2^2 + C \sum_{n=1}^{N} \epsilon_n \; - \sum_{n=1}^{N} \beta_n \epsilon_n \\ & - \sum_{n=1}^{N} \alpha_n (y_n ((\sum_{m=1}^{N} \alpha_m y_m \mathbf{x}_m) \mathbf{x}_n^T + b) + \epsilon_n - 1)\end{aligned}$$

$$\begin{aligned}\mathcal{L}(\mathbf{w}, b, \epsilon, \alpha, \beta) =& \frac{1}{2} \sum_{n=1}^{N} \sum_{m=1}^{N} \alpha_n \alpha_m y_n y_m \mathbf{x}_n \mathbf{x}_m^T + C \sum_{n=1}^{N} \epsilon_n \; - \sum_{n=1}^{N} \beta_n \epsilon_n \\ & - \sum_{n=1}^{N} \sum_{m=1}^{N} \alpha_n \alpha_m y_n y_m \mathbf{x}_m \mathbf{x}_n^T \; - \sum_{n=1}^{N} \alpha_n (y_n b + \epsilon_n - 1)\end{aligned}$$

Simplifying I

$$\begin{aligned}\mathcal{L}(\mathbf{w}, b, \epsilon, \alpha, \beta) =& \frac{1}{2}\sum_{n=1}^{N}\sum_{m=1}^{N}\alpha_n\alpha_m y_n y_m \mathbf{x}_n\mathbf{x}_m^T + C\sum_{n=1}^{N}\epsilon_n - \sum_{n=1}^{N}\beta_n\epsilon_n \\ & - \sum_{n=1}^{N}\sum_{m=1}^{N}\alpha_n\alpha_m y_n y_m \mathbf{x}_m\mathbf{x}_n^T - \sum_{n=1}^{N}\alpha_n(y_n b + \epsilon_n - 1)\end{aligned}$$

$$\begin{aligned}\mathcal{L}(\mathbf{w}, b, \epsilon, \alpha, \beta) = & -\frac{1}{2}\sum_{n=1}^{N}\sum_{m=1}^{N}\alpha_n\alpha_m y_n y_m \mathbf{x}_n\mathbf{x}_m^T + \sum_{n=1}^{N}(\alpha_n + \beta_n)\epsilon_n \\ & - \sum_{n=1}^{N}\beta_n\epsilon_n - \sum_{n=1}^{N}\alpha_n(y_n b + \epsilon_n - 1)\end{aligned}$$

Simplifying II

$$\mathcal{L}(\mathbf{w}, b, \epsilon, \alpha, \beta) = -\frac{1}{2}\sum_{n=1}^{N}\sum_{m=1}^{N}\alpha_n\alpha_m y_n y_m \mathbf{x}_n \mathbf{x}_m^T + \sum_{n=1}^{N}(\alpha_n + \beta_n)\epsilon_n \\ - \sum_{n=1}^{N}\beta_n\epsilon_n \; - \sum_{n=1}^{N}\alpha_n(y_n b + \epsilon_n - 1)$$

$$\mathcal{L}(\mathbf{w}, b, \epsilon, \alpha, \beta) = -\frac{1}{2}\sum_{n=1}^{N}\sum_{m=1}^{N}\alpha_n\alpha_m y_n y_m \mathbf{x}_n \mathbf{x}_m^T \; + \sum_{n=1}^{N}\alpha_n$$

Constraint Simplification

$$\begin{aligned} \alpha_n &\geq 0 \\ \beta_n &\geq 0 \\ \sum_{n=1}^{N} \alpha_n y_n &= 0 \\ \alpha_n + \beta_n &= C \end{aligned}$$

$$\begin{aligned} \sum_{n=1}^{N} \alpha_n y_n &= 0 \\ 0 \leq \alpha_n &\leq C \end{aligned}$$

The SVC Dual Optimization Problem

- The SVC dual optimization problem is given below where we have used α_n as the Lagrange multipliers.

$$\begin{aligned}
\arg\max_{\boldsymbol{\alpha}} \sum_{n=1}^{N} \alpha_n - \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} y_i y_j \alpha_i \alpha_j \mathbf{K}_{ij} \\
\text{s.t. } \forall n \;\; 0 \leq \alpha_n \leq C \\
\sum_{n=1}^{N} \alpha_n y_n = 0
\end{aligned}$$

- The matrix $\mathbf{K}$ that appears in the objective contains the inner products between all pairs of training data vectors: $\mathbf{K}_{ij} = \mathbf{x}_i \mathbf{x}_j^T$. The matrix $\mathbf{K}$ is thus positive semi-definite and the optimization problem is maximizing a concave function.

SVC Dual Predictions

- Given the value of $\boldsymbol{\alpha}$, we can make predictions as follows:

$$f_{svm}(\mathbf{x}) = \sum_{n=1}^{N} y_n \alpha_n \mathbf{x}_n \mathbf{x}^T + b$$

- There are a number of ways to derive the optimal b from the KKT conditions and an optimal $\boldsymbol{\alpha}$. One approach is to use the equation below where $\mathcal{S}$ is the set of data points satisfying $0 < \alpha_n < C$:

$$b = \frac{1}{|\mathcal{S}|} \sum_{n \in \mathcal{S}} \left(y_n - \sum_{m \in \mathcal{S}} y_m \alpha_m \mathbf{x}_m \mathbf{x}_n^T \right)$$

Support Vector Property

- The above results show that only the data points for which $\alpha_n > 0$ actually participate in defining the learned model and making predictions.
- The data points for which $\alpha_n > 0$ are the support vectors. For the linearly separable case, these are the data cases on the margin. For the non-separable case, the are the data points on, within and on the wrong side of the margin.

Basis Expansion

- One of the important applications of the SVC dual problem is constructing non-linear classifiers.
- Recall that we can make a linear model non-linear by introducing a basis function expansion $\phi(\mathbf{x})$.
- Plugging this into the dual, we obtain the learning problem shown below where the only change is $\mathbf{K}_{ij} = \phi(\mathbf{x}_i)\phi(\mathbf{x}_j)^T$.

$$\begin{aligned} \arg\max_{\boldsymbol{\alpha}} & \sum_{n=1}^{N} \alpha_n - \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} y_i y_j \alpha_i \alpha_j \mathbf{K}_{ij} \\ \text{s.t. } \forall n \;\; & 0 \leq \alpha_n \leq C \\ & \sum_{n=1}^{N} \alpha_n y_n = 0 \end{aligned}$$

Kernels

- The only requirement for the optimization problem to be well-defined is for the matrix $\mathbf{K}$ to be positive semi-definite.

- An alternative to using an explicit basis expansion is to use a kernel function $\mathcal{K}(\mathbf{x}, \mathbf{x}')$ that is guaranteed to generate a positive definite matrix $\mathbf{K}$ given any collection of data vectors $\mathbf{x}_n$. Such a kernel function is referred to as a *Mercer* kernel.

- In the learning problem, instead of setting $\mathbf{K}_{ij} = \phi(\mathbf{x}_i)\phi(\mathbf{x}_j)^T$, we set $\mathbf{K}_{ij} = \mathcal{K}(\mathbf{x}_i, \mathbf{x}_j)$.

- The prediction function becomes:

$$f_{svm}(\mathbf{x}) = \sum_{n=1}^{N} y_n \alpha_n \mathcal{K}(\mathbf{x}_n, \mathbf{x}) + b$$

Example: Kernels

- Linear Kernel: $\mathcal{K}(\mathbf{x}', \mathbf{x}) = \mathbf{x}'\mathbf{x}^T$
- Polynomial Kernel: $\mathcal{K}(\mathbf{x}', \mathbf{x}) = (1 + \mathbf{x}'\mathbf{x}^T)^B$
- Exponential (RBF) Kernel: $\mathcal{K}(\mathbf{x}', \mathbf{x}) = a \exp\left(-b\|\mathbf{x}' - \mathbf{x}\|_2^2 \right)$
- ... and many more kernel functions, including for inputs that are not real-valued.

Properties of Duals and Kernels

- The optimization problem remains concave when using kernels, but the decision function can be complex and non-linear.
- When there existis a kernel implementing a given basis expansion, the dual formulation results in a QP that scales with N while the primal results in a QP that scales with the size of the basis expansion.
- There exist kernel functions (like the RBF kernel) that do not correspond to finite-dimensional basis expansions that can be used exactly in the dual, but must be approximated to use in the primal.

Generalizing Kernelization

- Any convex supervised learning optimization problem with convex (or no constraints) has a similar dual to SVC and can usually be kernelized as well (linear regression, SVR, logistic regression, GLMs, etc.).
- However, only SVC and SVR have a direct sparsity property where we typically expect many dual variables to be zero.