Discussion of the paper “Linear Programming Boosting via Column Generation” by AYHAN Demiriz etc.

JUN Liao
Mar,13\textsuperscript{th},2003

1. Duality Theory

a. Define dual function

( from [Boyd 2002] )

5.1.1 The Lagrangian

We consider an optimization problem in the standard form (4.1):

\[
\begin{align*}
\text{minimize} & \quad f_0(x) \\
\text{subject to} & \quad f_i(x) \leq 0, \quad i = 1, \ldots, m \\
& \quad h_i(x) = 0, \quad i = 1, \ldots, p,
\end{align*}
\]

with variable $x \in \mathbb{R}^n$. We assume its domain $D = \bigcap_{i=0}^m \text{dom} f_i \cap \bigcap_{i=1}^p \text{dom} h_i$ is nonempty, and denote the optimal value of (5.1) by $p^*$. We do not assume the problem (5.1) is convex.

The basic idea in Lagrangian duality is to take the constraints in (5.1) into account by augmenting the objective function with a weighted sum of the constraint functions. We define the Lagrangian $L : \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^p \to \mathbb{R}$ associated with the problem (5.1) as

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

with $\text{dom} L = D \times \mathbb{R}^m \times \mathbb{R}^p$. We refer to $\lambda_i$ as the Lagrange multiplier associated with the $i$th inequality constraint $f_i(x) \leq 0$; similarly we refer to $\nu_i$ as the Lagrange multiplier associated with the $i$th equality constraint $h_i(x) = 0$. The vectors $\lambda$ and $\nu$ are called the dual variables or Lagrange multiplier vectors associated with the problem (5.1).

5.1.2 The Lagrange dual function

We define the Lagrange dual function (or just dual function) $g : \mathbb{R}^m \times \mathbb{R}^p \to \mathbb{R}$ as the minimum value of the Lagrangian over $x$: for $\lambda \in \mathbb{R}^m$, $\nu \in \mathbb{R}^p$,

\[
g(\lambda, \nu) = \inf_{x \in D} L(x, \lambda, \nu) = \inf_{x \in D} \left( f_0(x) + \sum_{i=1}^m \lambda_i f_i(x) + \sum_{i=1}^p \nu_i h_i(x) \right).
\]

When the Lagrangian is unbounded below in $x$, the dual function takes on the value $-\infty$. Since the dual function is the pointwise infimum of a family of affine functions of $(\lambda, \nu)$, it is concave, even when the problem (5.1) is not convex.
5.1.3 Lower bounds on optimal value

The dual function yields lower bounds on the optimal value \( p^* \) of the problem (5.1): for any \( \lambda \geq 0 \) and any \( \nu \) we have

\[
g(\lambda, \nu) \leq p^*. \tag{5.2}
\]

This important property is easily verified. Suppose \( \bar{x} \) is a feasible point for the problem (5.1), i.e., \( f_i(\bar{x}) \leq 0 \) and \( h_i(\bar{x}) = 0 \), and \( \lambda \geq 0 \). Then we have

\[
\sum_{i=1}^m \lambda_i f_i(\bar{x}) + \sum_{k=1}^p \nu_k h_k(\bar{x}) \leq 0,
\]

since each term in the first sum is nonpositive, and each term in the second sum is zero, and therefore

\[
L(\bar{x}, \lambda, \nu) = f_0(\bar{x}) + \sum_{i=1}^m \lambda_i f_i(\bar{x}) + \sum_{k=1}^p \nu_k h_k(\bar{x}) \leq f_0(\bar{x}).
\]

Hence

\[
g(\lambda, \nu) = \inf_{x \in D} L(x, \lambda, \nu) \leq L(\bar{x}, \lambda, \nu) \leq f_0(\bar{x}).
\]

Since \( g(\lambda, \nu) \leq f_0(\bar{x}) \) holds for every feasible point \( \bar{x} \), the inequality (5.2) follows. The lower bound (5.2) is illustrated in figure 5.1, for a simple problem with \( x \in \mathbb{R} \) and one inequality constraint.

The inequality (5.2) holds, but is vacuous, when \( g(\lambda, \nu) = -\infty \). The dual function gives a nontrivial lower bound on \( p^* \) only when \( \lambda \geq 0 \) and \( (\lambda, \nu) \in \text{dom } g \), i.e., \( g(\lambda, \nu) > -\infty \). We refer to a pair \((\lambda, \nu)\) with \( \lambda \geq 0 \) and \((\lambda, \nu) \in \text{dom } g \) as dual feasible, for reasons that will become clear later.

b. Construct dual problem

b1: Example: LP problem

\[
\begin{align*}
\text{minimize} & \quad C^T x \\
\text{s.t.} & \quad Ax \geq b \\
& \quad x \geq 0 
\end{align*}
\]

Lagrangian function:

\[
L(x,u,v) = C^T x + u^T (b - Ax) + v^T (-x) \\
= u^T b + (C^T - u^T A - v^T) x \\
\]

\[
x \geq 0, u, v \geq 0
\]

Since linear function is bounded below only when it is zero, so the dual function is
\[ g(u, v) = \inf_{x} L(x, u, v) = u^T b \quad \text{if} \quad C^T - u^T A - v^T = 0 \]
\[ = -\infty \quad \text{otherwise} \]
\[ u, v \geq 0 \]

Since we need to maximize \( g(u, v) \) to get to the optimal value \( f(\tilde{x}) = C^T \tilde{x} \), the dual problem is:

\[
\begin{align*}
\text{maximize} & \quad u^T b \\
\text{s.t.} & \quad C^T - u^T A = v^T \\
& \quad u, v \geq 0
\end{align*}
\]

Since \( v \geq 0 \), we can drop \( v \) and get the following equivalent formulation:

\[
\begin{align*}
\text{maximize} & \quad u^T b \\
\text{s.t.} & \quad C^T - u^T A \geq 0 \\
& \quad u \geq 0
\end{align*}
\]

which is:

\[
\begin{align*}
\text{maximize} & \quad b^T u \\
\text{s.t.} & \quad A^T u \leq C \\
& \quad u \geq 0
\end{align*}
\]

We can see that the dual form is a symmetric form of the primal form:

<table>
<thead>
<tr>
<th>Primal</th>
<th>Dual</th>
</tr>
</thead>
<tbody>
<tr>
<td>minimize ( C^T x )</td>
<td>maximize ( b^T u )</td>
</tr>
<tr>
<td>s.t. ( Ax \geq b )</td>
<td>s.t. ( A^T u \leq C )</td>
</tr>
<tr>
<td>( x \geq 0 )</td>
<td>( u \geq 0 )</td>
</tr>
</tbody>
</table>

Applying the above to the LP forms in the paper, we can easily derive the dual form, for example, from (2) to (3):
Let the matrix $H$ be a $\ell$ by $m$ matrix of all the possible labelings of the training data using functions from $\mathcal{H}$. Specifically $H_{ij} = h_j(x_i)$ is the label ($1$ or $-1$) given by weak hypothesis $h_j \in \mathcal{H}$ on the training point $x_i$. Each column $H_j$ of the matrix $H$ constitutes the output of weak hypothesis $h_j$ on the training data, while each row $H_i$ gives the outputs of all the weak hypotheses on the example $x_i$. There may be up to $2^\ell$ distinct weak hypotheses.

The following linear program can be used to minimize the quantity in Eq. (1):

$$
\begin{align*}
\min_{a, \xi} & \quad \sum_{i=1}^{m} a_i + C \sum_{i=1}^{\ell} \xi_i \\
\text{s.t.} & \quad y_i H_i a + \xi_i \geq 1, \quad \xi_i \geq 0, \quad i = 1, \ldots, \ell \\
& \quad a_i \geq 0, \quad i = 1, \ldots, m
\end{align*}
$$

(2)

where $C > 0$ is the tradeoff parameter between misclassification error and margin maximization. The dual of LP (2) is

$$
\begin{align*}
\max_u & \quad \sum_{i=1}^{\ell} u_i \\
\text{s.t.} & \quad \sum_{i=1}^{\ell} a_i y_i H_{ij} \leq 1, \quad j = 1, \ldots, m \\
& \quad 0 \leq u_i \leq C, \quad i = 1, \ldots, \ell
\end{align*}
$$

(3)

$(a_1, a_2, \ldots, a_\ell, \xi_1, \ldots, \xi_\ell)$ are the variables "x" in our case.

$(1, 1, \ldots, 1, C, \ldots, C)$ can be seen as "$C^T$" in our case,

$(1, 1, 1, \ldots, 1, (1) s)$ can be seen as "b" in our case

$[ y \ H \ 1 ]$ can be seen as "$A$" in our case
b2: Example: SVM (QP problem)

SVM primal form [Weston 1999]:

\[
\phi(w, \xi, b) = \frac{1}{2} (w \cdot w) + C \sum_{i=1}^{\ell} \xi_i \tag{2.20}
\]

subject to

\[
y_i((w \cdot x_i) + b) \geq 1 - \xi_i, \quad i = 1, \ldots, \ell, \tag{2.21}
\]

\[
\xi \geq 0. \tag{2.22}
\]

How to construct dual form:

The solution to this problem can be found by finding the saddle point of the Lagrange functional:

\[
\mathcal{L}(w, b, \xi, \alpha, \beta) = \frac{1}{2} (w \cdot w) + C \sum_{i=1}^{\ell} \xi_i - \sum_{i=1}^{\ell} \alpha_i (y_i(w \cdot x_i) + b y_i - 1 - \xi_i) - \sum_{i=1}^{\ell} \beta_i \xi_i \tag{2.24}
\]

where \(\alpha_i\) are the Lagrange multipliers. The Lagrangian has to be minimised with respect to \(w, b, \) and \(\xi_i\), and maximised with respect to \(\alpha_i \geq 0, \beta_i \geq 0\).

In the saddle point the solution \(L^* = \mathcal{L}(w^*, b^*, \xi^*, \alpha^*, \beta^*)\) satisfies

\[
\frac{\partial}{\partial b} \mathcal{L}^* = 0, \quad \frac{\partial}{\partial \xi_i} \mathcal{L}^* = 0, \quad \frac{\partial}{\partial w} \mathcal{L}^* = 0 \tag{2.25}
\]

which leads to the following conditions

\[
\sum_{i=1}^{\ell} \alpha_i^* y_i = 0, \tag{2.26}
\]

\[
\alpha_i^* + \beta_i^* = C, \quad 0 \leq \alpha_i^* \leq C, \quad i = 1, \ldots, \ell, \tag{2.27}
\]

\[
w^* = \sum_{i=1}^{\ell} \alpha_i^* y_i x_i. \tag{2.28}
\]
Substituting these results into the Lagrangian, the following optimization problem is obtained: to maximize the quadratic form (2.29) under constraints (2.30) and (2.31).

\[
W(\alpha) = \sum_{i=1}^{\ell} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{\ell} y_i y_j \alpha_i \alpha_j (x_i \cdot x_j) \tag{2.29}
\]

subject to

\[
0 \leq \alpha_i \leq C, \quad i = 1, \ldots, \ell, \tag{2.30}
\]

\[
\sum_{i=1}^{\ell} \alpha_i y_i = 0. \tag{2.31}
\]

The corresponding decision function can be derived by transforming (2.23) with (2.28) to give

\[
f(x, \alpha^*) = \text{sign} \left( \sum_{i=1}^{\ell} y_i \alpha_i^* (x_i \cdot x) + b^* \right). \tag{2.32}
\]

According to the classical Kuhn-Tucker theorem, at the saddle point the conditions

\[
\alpha_i^* \left\{ \left[ (x_i \cdot w^*) + b^* \right] y_i - 1 - \xi_i^* \right\} = 0, \quad i = 1, \ldots, \ell \tag{2.33}
\]

and

\[
\beta \xi_i = 0 \tag{2.34}
\]

hold true. From (2.27) we can deduce that if \( \alpha_i^* < C \) then \( \xi_i^* = 0 \). Thus, from (2.33) for any vector \( x_i \) if \( 0 < \alpha_i^* < C \) (\( \alpha_i^* \) is strictly between the bounds) then \( \left[ (x_i \cdot w^*) + b^* \right] y_i - 1 = 0 \) holds true. The threshold \( b \) can thus be calculated in the following way:

\[
b^* = \frac{1}{2} \left[ (w^* \cdot x^+) + (w^* \cdot x^-) \right] \tag{2.35}
\]

where \( x^+ \) is any vector belonging to the first class where \( 0 < \alpha_i^+ < C \) and \( x^- \) is any vector belonging to the second class, again, where \( 0 < \alpha_i^- < C \).

Duality theory also includes the strong duality condition – when the optimal value of dual problem equal to the optimal value of primal problem.
2. Boosting LP for classification

Assume there are $m$ hypotheses and $l$ training examples, the classifications of these $l$ training examples by the $m$ hypotheses form a $l \times m$ matrix $(H)$, the elements of this matrix are either +1 or −1, which represent correct classification or misclassification accordingly.

Boosting try to find a nonnegative linear combination of these hypotheses in order to make the final hypothesis being consistent with all the training examples. Some people found out that increasing margin after the final hypothesis is consistent can improve performance and there exists some generalization bounds on the margin. So the idea is: why not directly maximize margin? This then leads to a linear programming problem:

$$\max_{\alpha, \rho} \rho$$

s.t. $y_i H a \geq \rho, \ i = 1, \ldots, l$

$$\sum_{i=1}^{m} a_i = 1;$$

$$a_j \geq 0, \ j = 1, \ldots, m$$

$y_i H a \ an the final hypothesis’s margin on examples $x_i$.

For this LP problem, there might not exist a feasible solution due to the nature of problem or noisy data. By using the idea of “soft margin” from SVM, we allow training errors. So we always get a solution. The “soft margin” version of LP Boosting is (from the paper (4)):

$$\max_{\alpha, \xi, \rho} \rho - D \sum_{i=1}^{\ell} \xi_i$$

s.t. $y_i H a + \xi_i \geq \rho, \ i = 1, \ldots, \ell$

$$\sum_{i=1}^{m} a_i = 1, \ \xi_i \geq 0, \ i = 1, \ldots, \ell$$

$$a_j \geq 0, \ j = 1, \ldots, m$$

The dual of this is (from the paper (5)):

$$\min_{u, \beta} \beta$$

s.t. $\sum_{i=1}^{\ell} u_i y_i H_{ij} \leq \beta, \ j = 1, \ldots, m$

$$\sum_{i=1}^{\ell} u_i = 1, \ 0 \leq u_i \leq D, \ i = 1, \ldots, \ell$$
Now the problem becomes the minimization of the maximum edge for all the hypotheses. The edge of hypothesis $j$ is $\sum_{i=1}^{l} u_{i,y_j} H_{ij}$ with respect to the distribution $u$ on training examples.
3. Column generation for LPBoost

It is very common that the hypothesis space is huge, even infinite, i.e. there are many columns in H in our case. It is difficult to solve LP on this huge H. Column generation, as a method to solve LP incrementally, does not require that the matrix H be explicitly available. At each iteration, only a subset of the columns H is used to determine the current solution. There are some ways to tell whether the current solution is optimal over the full set H. If it is not optimal, a new column from H is put into H and a new LP is solved on the new H. It seems that it is very expensive to do this; but if the old solution of H is saved and used for solving the expanded H, there is very little computation effort to get a solution (using simplex method). The iteration continues until the solution is optimal for all the hypotheses. Even it may not reach optimal after some iterations, the approximation solution is still useful.

For the LP problem of maximizing the margin (paper (4),(5) ), we have special properties:

At a certain iteration, we have H, which is a subset of the columns of the full H. The LP primal solution ($\hat{a}, \hat{\rho}, \hat{\xi}$) and dual solution ($\hat{a}, \hat{\beta}$) satisfy the following conditions:

(From the paper, with modification: H is replaced by H )

the choice of parameters in the model and the performance of the eventual algorithm. The optimality conditions (Nash & Sofer, 1996) of LP (4) are primal feasibility:

$$y_i H \hat{a} + \xi_i \geq \rho, \quad i = 1, \ldots, \ell$$

$$\sum_{i=1}^{m} a_i = 1, \quad \xi_i \geq 0, \quad i = 1, \ldots, \ell$$

$$\hat{a}_i \geq 0, \quad i = 1, \ldots, m$$

(6)

dual feasibility:

$$\sum_{i=1}^{e} u_i y_i H_{ij} \leq \beta, \quad j = 1, \ldots, m$$

$$\sum_{i=1}^{e} u_i = 1, \quad 0 \leq u_i \leq D, \quad i = 1, \ldots, \ell$$

(7)

and complementarity, here stated as equality of the primal and dual objectives:

$$\rho - D \sum_{i=1}^{\ell} \xi_i = \beta$$

(8)
In the primal problem, the solution \((\hat{\alpha}, \hat{\beta}, \hat{\xi})\) is always feasible for all the hypotheses (satisfies conditions (6) with \(H\) replaced by \(H\)). The weights for all the hypotheses \(H\) is \(\alpha = (\hat{\alpha}, 0, 0, 0, \ldots)\) i.e., the hypotheses that are not used have weights 0. If the dual solution \((\hat{\alpha}, \hat{\beta})\) is also feasible for all the hypotheses (satisfies conditions (7) with \(H\) replaced by \(H\)), then we are done. The current solution is the optimal solution for all the hypotheses \(H\). If not, there exists a hypothesis \(H_j\) (a column in \(H\)) that violates condition

\[
(7): \sum_{i=1}^{\ell} u_i y_i H_{ij} > \hat{\beta}
\]

Thus we put this column \(H_j\) into \(H\) and solve LP on the new \(H\) again. Normally, we want to add the column \(j\) with the largest edge

\[
j: \sum_{i=1}^{\ell} u_i y_i H_{ij} = \max \sum_{i=1}^{\ell} u_i y_i H_{ij}
\]

so it can converge faster.

This gives us the LPBoost algorithm (algorithm 4.1 in the paper). Note the distribution \(u\) on training examples is used for finding the best hypothesis (the best column in \(H\)), this is similar to Boosting.

**Algorithm 4.1 (LPBoost).**

Given as input training set: \(S\)

\[
m \leftarrow 0 \quad \text{No weak hypotheses}
\]

\[
\alpha \leftarrow 0 \quad \text{All coefficients are 0}
\]

\[
\beta \leftarrow 0
\]

\[
u \leftarrow (\frac{1}{\ell}, \ldots, \frac{1}{\ell}) \quad \text{Corresponding optimal dual}
\]

**REPEAT**

\[
m \leftarrow m + 1
\]

Find weak hypothesis using equation (10):

\[
h_m \leftarrow \mathcal{H}(S, u)
\]

Check for optimal solution:

If \(\sum_{i=1}^{\ell} u_i y_i h_m(x_i) \leq \beta, m \leftarrow m - 1,\) break

\[
H_{im} \leftarrow h_m(x_i)
\]

Solve restricted master for new costs:

\[
\arg \min \beta
\]

s.t. \(\sum_{i=1}^{\ell} u_i y_i h_j(x_i) \leq \beta\)

\[
(j, \beta) \leftarrow j = 1, \ldots, m
\]

\[
\sum_{i=1}^{\ell} u_i = 1
\]

\[
0 \leq u_i \leq D, \ i = 1, \ldots, \ell
\]

**END**

\[
a \leftarrow \text{Lagrangian multipliers from last LP}
\]

return \(m, f = \sum_{j=1}^{m} \hat{a}_j h_j\)
4. Total corrective update property for LPBoosting by column generation.

One interesting thing about LPBoost by column generation is: the returned weak hypothesis (the column to be added into H) is always new – a hypothesis that has not been returned before, otherwise the algorithm stops and the optimal solution is found. This can be easily verified as follows:

At certain iteration, all the weak hypotheses returned so far (H) have edges $\leq \hat{\beta}$ given distribution $\hat{u}$ on training examples. Since weak learner always finds the weak hypothesis with the largest edge given $\hat{u}$, only two things can happen:

a. A weak hypotheses with edge $> \hat{\beta}$ is returned. This hypothesis should be new because all the weak hypotheses returned before (H) have edges $\leq \hat{\beta}$ right now.

b. A weak hypothesis with edge $\leq \hat{\beta}$ is returned. This means all the hypotheses’ edges $\leq \hat{\beta}$. Thus the dual feasibility is satisfied and the algorithm stops. The current solution is the optimal solution for all the hypotheses.

So a new weak hypothesis is always returned at each iteration. This property is useful for feature selection.

Notice that in LPBoost, the weights for all weak hypotheses selected so far are updated at each iteration depending on the LP solution. While in AdaBoost, once a weight for a selected weak hypothesis is generated, it is fixed.