Duality.

Seminar

Optimization for ML. Faculty of Computer Science. HSE University

Dual function

The **general mathematical programming problem** with functional constraints:

$$
\begin{split}
& f_0(x) \to \min_{x \in \mathbb{R}^n}\\
\text{s.t. } & f_i(x) \leq 0, \; i = 1, \ldots, m\\
& h_i(x) = 0, \; i = 1, \ldots, p
\end{split}
$$

And the Lagrangian, associated with this problem:

$$
L(x, \lambda, \nu) = f_0(x) + \sum\limits_{i=1}^m \lambda_i f_i(x) + \sum\limits_{i=1}^p\nu_i h_i(x) = f_0(x) + \lambda^\top f(x) + \nu^\top h(x)
$$

We assume $\mathcal{D} = \bigcap\limits_{i=0}^m\textbf{dom } f_i \cap \bigcap\limits_{i=1}^p\textbf{dom } h_i$ is nonempty. 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 $\lambda \in \mathbb{R}^m, \nu \in \mathbb{R}^p$

$$
g(\lambda, \nu) = \inf_{x\in \mathcal{D}} L(x, \lambda, \nu) = \inf_{x\in \mathcal{D}} \left(f_0(x) +\sum\limits_{i=1}^m \lambda_i f_i(x) + \sum\limits_{i=1}^p\nu_i h_i(x) \right)
$$

Dual function. Summary

Primal

Function:

$$f_0(x)$$

Variables:

$$x \in S \subseteq \mathbb{R}^{\ltimes}$$

Constraints:

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

$$h_i(x) = 0, \ i = 1, \ldots, p$$

Dual

Function:

$$g(\lambda, \nu) = \min_{x \in \mathcal{D}} L(x, \lambda, \nu)$$

Variables

$$\lambda \in \mathbb{R}^m_+, \nu \in \mathbb{R}^p$$

Constraints:

$$\lambda_i \geq 0, \forall i \in \overline{1, m}$$

Strong Duality

It is common to name this relation between optimals of primal and dual problems as **weak duality**. For problem, we have:

$$
d^* \leq p^*
$$

While the difference between them is often called **duality gap:**

$$
0 \leq p^* - d^*
$$

Strong duality happens if duality gap is zero:

$$
p^* = d^*
$$

Slater's condition

If for a convex optimization problem (i.e., assuming minimization, f_0, f_i are convex and h_i are affine), there exists a point x such that $h(x) = 0$ and $f_i(x) < 0$ (existance of a **strictly feasible point**), then we have a zero duality gap and KKT conditions become necessary and sufficient.

Reminder of KKT statements

Suppose we have a **general optimization problem**

$$
\begin{split}
& f_0(x) \to \min_{x \in \mathbb{R}^n} \\
\text{s.t. } & f_i(x) \leq 0, \; i = 1, \ldots, m \\
& h_i(x) = 0, \; i = 1, \ldots, p
\end{split}
\tag{1}
$$

and **convex optimization problem**, where all equality constraints are affine:

$$
h_i(x) = a_i^T x - b_i, i \in 1, \ldots p.
$$

The **KKT system** is:

$$
\begin{split}
& \nabla_x L(x^*, \lambda^*, \nu^*) = 0 \\
& \nabla_\nu L(x^*, \lambda^*, \nu^*) = 0 \\
& \lambda_i^* \geq 0, i = 1, \ldots, m \\
& \lambda_i^* f_i(x^*) = 0, i = 1, \ldots, m \\
& f_i(x^*) \leq 0, i = 1, \ldots, m
\end{split}
\tag{2}
$$

KKT becomes necessary

If x^* is a solution of the original problem Equation 1, then if any of the following regularity conditions is satisfied:

- **Strong duality** If $f_1, \ldots f_m, h_1, \ldots h_p$ are differentiable functions and we have a problem Equation 1 with zero duality gap, then Equation 2 are necessary (i.e. any optimal set x^*, λ^*, ν^* should satisfy Equation 2)
- **LCQ** (Linearity constraint qualification). If $f_1, \ldots f_m, h_1, \ldots h_p$ are affine functions, then no other condition is needed.
- **LICQ** (Linear independence constraint qualification). The gradients of the active inequality constraints and the gradients of the equality constraints are linearly independent at x^*
- **SC** (Slater's condition) For a convex optimization problem (i.e., assuming minimization, f_i are convex and h_j is affine), there exists a point x such that $h_j(x) = 0$ and $g_i(x) < 0$.

Than it should satisfy Equation 2

KKT in convex case

If a convex optimization problem with differentiable objective and constraint functions satisfies Slater's condition, then the KKT conditions provide necessary and sufficient conditions for optimality: Slater's condition implies that the optimal duality gap is zero and the dual optimum is attained, so x^* is optimal if and only if there are (λ^*, ν^*) that, together with x^*, satisfy the KKT conditions.

Problem 1. Dual LP

Question

Ensure, that the following standard form *Linear Programming* (LP):

$$
\begin{split}
&\min_{x \in \mathbb{R}^n} c^\top x \\
\text{s.t. } & Ax = b \\
& x_i \geq 0, \; i = 1, \dots, n
\end{split}
$$

Has the following dual:

$$
\begin{split}
&\max_{y \in \mathbb{R}^n} b^\top y \\
\text{s.t. } & A^T y \preceq c
\end{split}
$$

Find the dual problem to the problem above (it should be the original LP).

Problem 2. Lagrange matrix multiplier

Question

Let matrices $X \in \mathbb{R}^{n \times m}$, $C \in \mathbb{R}^{n \times m}$, $A \in \mathbb{R}^{k \times n}$, $B \in \mathbb{R}^{k \times m}$. Setting the task:

$$f(X) = \langle C, X \rangle \longrightarrow \min_X$$

$$\text{s.t } AX \leqslant B$$

Find the dual problem to the problem above.

Problem 3. Projection onto probability simplex

Question

Find the Euclidean projection of $x \in \mathbb{R}^n$ onto probability simplex

$$\Delta = \{z \in \mathbb{R}^n \mid z \succeq 0, \mathbf{1}^\top z = 1\},$$

i.e. solve the following problem:

$$x^* = P_\Delta(y) = \operatorname*{argmin}_{x \in \mathbb{R}^n_+} \frac{1}{2}\|x - y\|_2^2$$

$$\text{s.t. } \mathbf{1}^\top x = 1$$

Problem 3 solution: using duality problem

The "partial" Lagrangian, considering only equality constraints:

$$
L(x, \nu) = \frac{1}{2}\|x - y\|_2^2 + \nu \left(\mathbf{1}^T x - 1\right)
$$

Problem 3 solution: using duality problem

The “partial” Lagrangian, considering only equality constraints:

$$
L(x,\nu) = \frac{1}{2}\|x-y\|_2^2 + \nu\left(\mathbf{1}^T x - 1\right)
$$

To find a solution (x^*, ν^*), let's set a saddle point problem:

$$
\operatorname*{argmax}_{\nu} L(x,\nu) \longrightarrow \min_{x \succeq 0}
$$

Problem 3 solution: using duality problem

The "partial" Lagrangian, considering only equality constraints:

$$L(x, \nu) = \frac{1}{2}\|x - y\|_2^2 + \nu\left(\mathbf{1}^T x - 1\right)$$

To find a solution (x^*, ν^*), let's set a saddle point problem:

$$\underset{\nu}{\operatorname{argmax}}\, L(x, \nu) \longrightarrow \min_{x \succeq 0}$$

We will solve this problem in two stages:

- We first solve $\underset{x \succeq 0}{\operatorname{argmin}} L(x, \nu)$ to get x^*
- Then we use x^* to get ν^* by solving $\underset{\nu}{\operatorname{argmax}} L(x^*, \nu)$

Problem 3 solution: using duality problem

1. Let's solve $\underset{x\succeq 0}{\text{argmin}} L(x,\nu)$:

$$\min_{x\succeq 0} L(x,\nu) = \min_{x\succeq 0}\left(\frac{1}{2}\|x-y\|_2^2 + \nu\left(\mathbf{1}^T x - 1\right)\right) = \min_{x\succeq 0}\left(\frac{1}{2}\|x-y\|_2^2 + \nu\mathbf{1}^T x\right)$$

Problem 3 solution: using duality problem

1. Let's solve $\underset{x\succeq 0}{\text{argmin}} L(x,\nu)$:

$$\min_{x\succeq 0} L(x,\nu) = \min_{x\succeq 0}\left(\frac{1}{2}\|x-y\|_2^2 + \nu\left(\mathbf{1}^T x - 1\right)\right) = \min_{x\succeq 0}\left(\frac{1}{2}\|x-y\|_2^2 + \nu\mathbf{1}^T x\right)$$

$$\min_{x\succeq 0}\left(\frac{1}{2}\|x-y\|_2^2 + \nu\mathbf{1}^T x\right) = \min_{x\succeq 0}\left(\sum_{i=1}^n \frac{1}{2}(x_i - y_i)^2 + \nu x_i\right) = \min_{x\succeq 0}\left(\sum_{i=1}^n l_i(x_i)\right)$$

Problem 3 solution: using duality problem

1. Let's solve $\underset{x \succeq 0}{\operatorname{argmin}} L(x, \nu)$:

$$\min_{x \succeq 0} L(x, \nu) = \min_{x \succeq 0} \left(\frac{1}{2}\|x - y\|_2^2 + \nu \left(\mathbf{1}^T x - 1 \right) \right) = \min_{x \succeq 0} \left(\frac{1}{2}\|x - y\|_2^2 + \nu \mathbf{1}^T x \right)$$

$$\min_{x \succeq 0} \left(\frac{1}{2}\|x - y\|_2^2 + \nu \mathbf{1}^T x \right) = \min_{x \succeq 0} \left(\sum_{i=1}^n \frac{1}{2}(x_i - y_i)^2 + \nu x_i \right) = \min_{x \succeq 0} \left(\sum_{i=1}^n l_i(x_i) \right)$$

L is minimized if all l_i are minimized, so we have scalar problem

$$l_i(x_i) = \frac{1}{2}(x_i - y_i)^2 + \nu x_i \longrightarrow \min_{x_i \geqslant 0}$$

Problem 3 solution: using duality problem

1. Let's solve $\underset{x \succeq 0}{\text{argmin}} L(x, \nu)$:

$$\min_{x \succeq 0} L(x, \nu) = \min_{x \succeq 0} \left(\frac{1}{2} \|x - y\|_2^2 + \nu \left(\mathbf{1}^T x - 1 \right) \right) = \min_{x \succeq 0} \left(\frac{1}{2} \|x - y\|_2^2 + \nu \mathbf{1}^T x \right)$$

$$\min_{x \succeq 0} \left(\frac{1}{2} \|x - y\|_2^2 + \nu \mathbf{1}^T x \right) = \min_{x \succeq 0} \left(\sum_{i=1}^{n} \frac{1}{2} (x_i - y_i)^2 + \nu x_i \right) = \min_{x \succeq 0} \left(\sum_{i=1}^{n} l_i(x_i) \right)$$

L is minimized if all l_i are minimized, so we have scalar problem

$$l_i(x_i) = \frac{1}{2} (x_i - y_i)^2 + \nu x_i \longrightarrow \min_{x_i \geqslant 0}$$

And the solution to this problem is

- $x_i^* = (y_i - \nu)$ if $y_i - \nu \geqslant 0$
- $x_i^* = 0$ if $y_i - \nu \leq 0$

Problem 3 solution: using duality problem

So, solution of the first subtask is

$$x^* = [y - \nu \mathbf{1}]_+$$

2. Now we must find ν. To do this, let's use the constraint:

Problem 3 solution: using duality problem

So, solution of the first subtask is

$$x^* = [y - \nu \mathbf{1}]_+$$

2. Now we must find ν. To do this, let's use the constraint:

$$\sum_{i=1}^{n} x_i^* = \sum_{i=1}^{n} [y_i - \nu]_+ = \sum_{i=1}^{n} max\{0, y_i - \nu\} = \sum_{j: y_j > \nu} (y_j - \nu) = 1$$

Problem 3 solution: using duality problem

So, solution of the first subtask is

$$x^* = [y - \nu \mathbf{1}]_+$$

2. Now we must find ν. To do this, let's use the constraint:

$$\sum_{i=1}^{n} x_i^* = \sum_{i=1}^{n} [y_i - \nu]_+ = \sum_{i=1}^{n} max\{0, y_i - \nu\} = \sum_{j: y_j > \nu} (y_j - \nu) = 1$$

In other words, in this sum, we discard those components of the y that are less than ν. To find ν, using the expression above, let's sort the components of the vector and present a set

$$\mathcal{J} = \{j : y_j > \nu\}, \quad |\mathcal{J}| = K,$$

where elemets of y already sorted: $y_1 \geqslant y_2 \geqslant ... \geqslant y_n$

Problem 3 solution: using duality problem

So we have

$$\sum_{j:y_j>\nu}(y_j-\nu)=\sum_{j\in\mathcal{J}}y_j-K\nu=1\Rightarrow\nu=\frac{\sum_{j\in\mathcal{J}}y_j-1}{K}$$

The final probability simplex projection algorithm includes 3 steps:

- Sort y
- Find K, which is the last integer in $\{1,2,...,n\}$ that $y_K-\frac{\sum_{j\in\mathcal{J}}y_j-1}{K}>0$
- Output $\nu=\frac{\sum_{j\in\mathcal{J}}y_j-1}{K}$ for $x=P_\Delta(y)=[y-\nu\mathbf{1}]_+$

Problem 3 solution: using duality problem

So we have

$$\sum_{j:y_j>\nu}(y_j-\nu)=\sum_{j\in\mathcal{J}}y_j-K\nu=1\Rightarrow\nu=\frac{\sum_{j\in\mathcal{J}}y_j-1}{K}$$

The final probability simplex projection algorithm includes 3 steps:

- Sort y
- Find K, which is the last integer in $\{1,2,...,n\}$ that $y_K-\frac{\sum_{j\in\mathcal{J}}y_j-1}{K}>0$
- Output $\nu=\frac{\sum_{j\in\mathcal{J}}y_j-1}{K}$ for $x=P_\Delta(y)=[y-\nu\mathbf{1}]_+$

The most expensive part here is step-1, using quick sort, the worst computational complexity is $\mathcal{O}(n\log n)$

Problem 3 solution: another algorithm $\mathcal{O}(n)$ (?)

Here is the formulation of the algorithm:

$$
\begin{array}{l}
\textsc{Input}\ \text{A vector } \mathbf{v} \in \mathbb{R}^n \text{ and a scalar } z > 0 \\
\textsc{Initialize}\ U = [n] \quad s = 0 \quad \rho = 0 \\
\textsc{While}\ U \neq \phi \\
\quad \textsc{Pick}\ k \in U \text{ at random} \\
\quad \textsc{Partition}\ U\text{:} \\
\qquad G = \{j \in U \mid v_j \geq v_k\} \\
\qquad L = \{j \in U \mid v_j < v_k\} \\
\quad \textsc{Calculate}\ \ \Delta\rho = |G| \ \ ; \ \ \Delta s = \sum_{j \in G} v_j \\
\quad \textsc{If}\ (s + \Delta s) - (\rho + \Delta\rho) v_k < z \\
\qquad s = s + \Delta s \ \ ; \ \ \rho = \rho + \Delta\rho \ \ ; \ \ U \leftarrow L \\
\quad \textsc{Else} \\
\qquad U \leftarrow G \setminus \{v_k\} \\
\quad \textsc{EndIf} \\
\textsc{Set}\ \theta = (s - z)/\rho \\
\textsc{Output}\ \mathbf{w} \text{ s.t. } v_i = \max\{v_i - \theta\ ,\ 0\}
\end{array}
$$

Figure 1: Linear time projection algorithm pseudo-code

Problem 3 solution: another algorithm $\mathcal{O}(n)$ (?)

In short, what is the difference between this algorithm and the first one? In the **step 1**.

- Algorithm 2 (pivot-algorithm) does not sort the entire array, but randomly selects a "pivot" and "slices" the list, similar to Quickselect (quick median search).
- On average, it gives $\mathcal{O}(n)$, but in the worst case (unsuccessful pivots), theoretically it can "fail" to $\mathcal{O}(n^2)$ (!)
- So, the statement about the difficulty of $\mathcal{O}(n)$ in the original article was a mistake. Article provides an attempt to fix this and an overview of the mistake.
- The code for comparing this algorithm with the previous one is here

Problem 4. Projection onto the unit simplex VS projection onto the l_1 ball

Projection onto the l_1 ball

The same article mentions the connection between searching for a projection on the unit simplex and on the l_1 ball. Previous problem:

$$x_1^* = \operatorname*{argmin}_{x \in \mathbb{R}^n} \frac{1}{2}\|x - y\|_2^2$$

$$\text{s.t. } \mathbf{1}^\top x = 1, \ x \succeq 0$$

New problem:

$$x_2^* = \operatorname*{argmin}_{x \in \mathbb{R}^n} \frac{1}{2}\|x - y\|_2^2$$

$$\text{s.t. } \|x\|_1 \leqslant 1$$

Let's show idea how to reduce the first to the second.

Problem 4. Projection onto the unit simplex VS projection onto the l_1 ball

1. If $\|y\|_1 \leqslant 1$ then you don't need to do anything: it's already inside (or on the border) l_1-ball, therefore, the desired projection is equal to the y

Problem 4. Projection onto the unit simplex VS projection onto the l_1 ball

1. If $\|y\|_1 \leqslant 1$ then you don't need to do anything: it's already inside (or on the border) l_1-ball, therefore, the desired projection is equal to the y
2. If $\|y\|_1 > 1$ then the optimum will be exactly on the border, that is, it must be fulfilled $\|y\|_1 = 1$

Problem 4. Projection onto the unit simplex VS projection onto the l_1 ball

1. If $\|y\|_1 \leqslant 1$ then you don't need to do anything: it's already inside (or on the border) l_1-ball, therefore, the desired projection is equal to the y
2. If $\|y\|_1 > 1$ then the optimum will be exactly on the border, that is, it must be fulfilled $\|y\|_1 = 1$
3. The following lemma is proved in the paper:

Lemma

In the optimal solution, each non-zero coordinate x_i must have the same sign as the y_i. Formally,

$$x_i \neq 0 \Rightarrow sign(x_i) = sign(y_i)$$

Problem 4. Projection onto the unit simplex VS projection onto the l_1 ball

4. Thanks to the previous paragraph, it is sufficient to consider the "modules" of coordinates. An auxiliary vector is introduced

$$u \in \mathbb{R}^n, u_i = |y_i|$$

Problem 4. Projection onto the unit simplex VS projection onto the l_1 ball

4. Thanks to the previous paragraph, it is sufficient to consider the "modules" of coordinates. An auxiliary vector is introduced

$$u \in \mathbb{R}^n, u_i = |y_i|$$

Then the constraint $\|x\|_1 \leqslant 1$ and the condition "sign x_i coincides with sign y_i" are equivalent to the problem

$$\min_{u \succeq 0} \|u - |y|\|_2^2 \text{ s.t. } \|u\|_1 = 1$$

Problem 4. Projection onto the unit simplex VS projection onto the l_1 ball

4. Thanks to the previous paragraph, it is sufficient to consider the "modules" of coordinates. An auxiliary vector is introduced

$$u \in \mathbb{R}^n, u_i = |y_i|$$

Then the constraint $\|x\|_1 \leqslant 1$ and the condition "sign x_i coincides with sign y_i" are equivalent to the problem

$$\min_{u \succeq 0} \|u - |y|\|_2^2 \text{ s.t. } \|u\|_1 = 1$$

But this is the problem of projection onto a probability simplex with a sum of coordinates equals to 1.

Problem 4. Projection onto the unit simplex VS projection onto the l_1 ball

4. Thanks to the previous paragraph, it is sufficient to consider the "modules" of coordinates. An auxiliary vector is introduced

$$u \in \mathbb{R}^n, u_i = |y_i|$$

Then the constraint $\|x\|_1 \leqslant 1$ and the condition "sign x_i coincides with sign y_i" are equivalent to the problem

$$\min_{u \succeq 0} \|u - |y|\|_2^2 \text{ s.t. } \|u\|_1 = 1$$

But this is the problem of projection onto a probability simplex with a sum of coordinates equals to 1.

Let's denote the found solution to the problem above for u^*. Then we return to the original x^*, restoring the signs:

$$\boxed{x_i^* = sign(y_i) \cdot u_i^*}$$

This x_i^* solution that is the desired projection onto the l_1 ball.

Problem 5. Dual to SVM