AIMS Course 3: Optimisation

Solutions to mathematical problems

Jake Levi

October 2022

1 Introduction

We consider optimisation problems consisting of a decision variable x, which takes values in a set $\mathcal{X}$ (referred to as the domain of x), an objective function $f_0(x)$, inequality constraints $f_i(x)$ (for $i \in \{1, \dots, m\}$), and equality constraints $h_i(x)$ (for $i \in \{1, \dots, p\}$), which can be expressed in the following form:

$$\begin{aligned} \underset{x \in \mathcal{X}}{\text{Minimise}} \quad & f_0(x) \\ \text{Subject to} \quad & f_i(x) \le 0 \quad i \in \{1, \dots, m\} \\ & h_i(x) = 0 \quad i \in \{1, \dots, p\} \end{aligned} \tag{1}$$

We refer to the set of values of $x \in \mathcal{X}$ which satisfy the equality and inequality constraints as the feasible set, which is denoted by $\mathcal{F}$:

$$\mathcal{F} = \{x \in \mathcal{X} : (\forall i \in \{1, \dots, m\}) \quad f_i(x) \le 0 \quad \text{and} \quad (\forall i \in \{1, \dots, p\}) \quad h_i(x) = 0\} \tag{2}$$

The limit of the smallest value of $f_0(x)$ for any value of x in the feasible set $\mathcal{F}$ is referred to as the optimal cost, and denoted by p^*:

$$p^* = \inf_{x \in \mathcal{F}} [f_0(x)] \tag{3}$$

When solving an optimisation problem in the form described by equation 1, it is useful to introduce the Lagrangian function [1] (intuition for the form of the Lagrangian function is provided in appendix A), which is a function of the decision variable $x \in \mathcal{X}$, and also the variables $\lambda \in \mathbb{R}^m$ and $\nu \in \mathbb{R}^p$, which are referred to as the Lagrange multipliers for the inequality and equality constraints respectively:

$$\mathcal{L}(x, \lambda, \nu) = f_0(x) + \sum_{i=1}^{m} [\lambda_i f_i(x)] + \sum_{i=1}^{p} [\nu_i h_i(x)] \tag{4}$$

The limit of the smallest value of the Lagrangian function for any value of x in its domain $\mathcal{X}$ (not only in the feasible set $\mathcal{F}$) as a function of the Lagrange multipliers λ and ν is referred to as the Lagrange dual function, denoted by g:

$$\begin{aligned} g(\lambda, \nu) &= \inf_{x \in \mathcal{X}} [\mathcal{L}(x, \lambda, \nu)] && (5) \\ &= \inf_{x \in \mathcal{X}} \left[f_0(x) + \sum_{i=1}^{m} [\lambda_i f_i(x)] + \sum_{i=1}^{p} [\nu_i h_i(x)] \right] && (6) \end{aligned}$$

The dual function is concave with respect to λ and ν, which is equivalent to the following statements (which are proved in appendix B):

$$(\forall \lambda^{(1)}, \lambda^{(2)} \in \mathbb{R}^m)(\forall \alpha \in [0, 1]) \quad g(\alpha\lambda^{(1)} + (1 - \alpha)\lambda^{(2)}, \nu) \ge \alpha g(\lambda^{(1)}, \nu) + (1 - \alpha) g(\lambda^{(2)}, \nu) \tag{7}$$

$$(\forall \nu^{(1)}, \nu^{(2)} \in \mathbb{R}^p)(\forall \alpha \in [0, 1]) \quad g(\lambda, \alpha\nu^{(1)} + (1 - \alpha)\nu^{(2)}) \ge \alpha g(\lambda, \nu^{(1)}) + (1 - \alpha) g(\lambda, \nu^{(2)}) \tag{8}$$

We also note that for any $\lambda \ge 0$ (where $\ge$ is understood to apply element-wise to each element of λ) and for any ν, the Lagrangian dual function $g(\lambda, \nu)$ is a lower bound for the optimal cost p^*. To prove this, it is useful to note that for any feasible point $\tilde{x} \in \mathcal{F}$ (which necessarily satisfies $f_i(x) \le 0$ for all $i \in \{1, \dots, m\}$ and $h_i(x) = 0$ for all

$i \in \{1, \ldots, p\}$), and for any $\lambda \geq 0$ and any ν, the following inequality holds:

$$(\forall \tilde{x} \in \mathcal{F}, \lambda \geq 0, \nu) \quad 0 \geq \sum_{i=1}^{m} [\underbrace{\lambda_i}_{\geq 0} \underbrace{f_i(\tilde{x})}_{\leq 0}] + \sum_{i=1}^{p} [\nu_i \underbrace{h_i(\tilde{x})}_{=0}] \tag{9}$$

$$\begin{aligned} \Rightarrow (\forall \tilde{x} \in \mathcal{F}, \lambda \geq 0, \nu) \quad f_0(\tilde{x}) &\geq f_0(\tilde{x}) + \sum_{i=1}^{m} [\lambda_i f_i(\tilde{x})] + \sum_{i=1}^{p} [\nu_i h_i(\tilde{x})] & (10) \\ &= \mathcal{L}(\tilde{x}, \lambda, \nu) & (11) \\ &\geq \inf_{x \in \mathcal{X}} [\mathcal{L}(x, \lambda, \nu)] & (12) \\ &= g(\lambda, \nu) & (13) \end{aligned}$$

Since the Lagrangian dual function is a lower bound on the objective function for any value of $\tilde{x}$ in the feasible set $\mathcal{F}$, it is also true for the value of $\tilde{x} \in \mathcal{F}$ which minimises the objective function, $f_0(x)$:

$$\begin{aligned} (\forall \tilde{x} \in \mathcal{F}, \lambda \geq 0, \nu) \quad g(\lambda, \nu) &\leq f_0(\tilde{x}) & (14) \\ \Rightarrow (\forall \lambda \geq 0, \nu) \quad g(\lambda, \nu) &\leq \inf_{x \in \mathcal{F}} [f_0(x)] & (15) \\ &= p^* & (16) \end{aligned}$$

The limit of the greatest value of $g(\lambda, \nu)$ for any $\lambda \geq 0$ and ν is the best lower bound on p^*, and is referred to as the dual optimal cost, denoted by d^*:

$$\begin{aligned} d^* &= \sup_{\lambda \geq 0, \nu} [g(\lambda, \nu)] & (17) \\ &= \sup_{\lambda \geq 0, \nu} \left[\inf_{x \in \mathcal{X}} [\mathcal{L}(x, \lambda, \nu)] \right] & (18) \\ &\leq p^* & (19) \end{aligned}$$

Therefore we have that the dual optimal cost d^* is always less than or equal to the optimal cost, p^* (also referred to as the primal optimal cost). An interesting symmetry exists between the primal optimal cost, p^* and the dual optimal cost, d^*. To see this, we start by noting that the Lagrangian function $\mathcal{L}(x, \lambda, \nu)$ is unbounded above (and below) with respect to $\lambda \geq 0$ and ν unless x is in the feasible set $\mathcal{F}$, in which case the Lagrangian function is bounded above by the objective function, $f_0(x)$:

$$\sup_{\lambda \geq 0, \nu} [\mathcal{L}(x, \lambda, \nu)] = \begin{cases} f_0(x) & x \in \mathcal{F} \\ \infty & \text{otherwise} \end{cases} \tag{20}$$

$$\begin{aligned} \Rightarrow \inf_{x \in \mathcal{X}} \left[\sup_{\lambda \geq 0, \nu} [\mathcal{L}(x, \lambda, \nu)] \right] &= \inf_{x \in \mathcal{F}} [f_0(x)] & (21) \\ &= p^* & (22) \\ \Rightarrow \sup_{\lambda \geq 0, \nu} \left[\inf_{x \in \mathcal{X}} [\mathcal{L}(x, \lambda, \nu)] \right] &\leq \inf_{x \in \mathcal{X}} \left[\sup_{\lambda \geq 0, \nu} [\mathcal{L}(x, \lambda, \nu)] \right] & (23) \end{aligned}$$

This latter inequality is a special case of the Max–min inequality, and in the context of optimisation, it is known as the principle of weak duality. In cases where equality holds, and the dual optimal cost is equal to the primal optimal cost, $d^* = p^*$, this is known as strong duality.

The Lagrangian and Lagrangian dual functions are therefore useful in constrained optimisation, because they allow us to minimise the Lagrangian function $\mathcal{L}$ with respect to x as a function of λ and ν without constraints (which yields the Langrangian dual function g), and then to maximise the Langrangian dual function g (which is necessarily concave) with respect to λ and ν, subject only to the constraints that each element of λ is nonnegative (if the original problem has no inequality constraints then there are no constraints on the maximisation of g at all), which yields the dual optimal cost d^*. The dual optimal cost provides a lower bound on the value p^* in all cases, and in the case of strong duality, provides the value of p^* itself.

2 Convex Sets, Functions And Problems

2.1 Standard Form Of Convex Problems

A convex optimisation problem is one in which the objective function $f_0(x)$ (defined in equation 1) is a convex function, and the feasible set $\mathcal{F}$ (defined in equation 2) is a convex set. In order for $\mathcal{F}$ to be a convex set, any inequality constraints must restrict x to be in a convex subset of its domain, and any equality constraints must restrict x to be in an affine subspace of its domain. Therefore, a convex optimisation problem in the form of equation 1 is said to be a standard form convex optimisation problem if the objective function $f_0(x)$ is a convex function, all inequality constraint functions $f_i(x)$ are convex functions ($\forall i \in \{1, \dots, m\}$), and all equality constraint functions $h_i(x)$ are affine functions ($\forall i \in \{1, \dots, p\}$), in which case the problem is a convex optimisation problem. However, not all convex optimisation problems are in standard form, as the following example demonstrates:

$$
\begin{aligned}
\underset{x \in \mathbb{R}^2}{\text{Minimise}} \quad & x_1^2 + x_2^2 \\
\text{Subject to} \quad & f_1(x) = \frac{x_1}{1 + x_2^2} \leq 0 \\
& h_1(x) = (x_1 + x_2)^2 = 0
\end{aligned}
\tag{24}
$$

In this case, $f_1(x)$ is not a convex function, and $h_1(x)$ is not an affine function, therefore the problem is not a standard form convex optimisation problem, however the objective function is a convex function, and the feasible set defined by these constraints is a convex set:

$$
\begin{aligned}
& \frac{x_1}{1 + x_2^2} \leq 0 \\
(\forall x_2 \in \mathbb{R}) \quad & \frac{1}{1 + x_2^2} > 0 \\
& \Rightarrow x_1 \leq 0 \\
& (x_1 + x_2)^2 = 0 \\
& \Rightarrow x_1 + x_2 = 0 \\
& \Rightarrow x_2 = -x_1 \\
& \Rightarrow x_2 \geq 0 \\
& \Rightarrow \mathcal{F} = \{x \in \mathbb{R}^2 : x_1 \leq 0 \quad \text{and} \quad x_2 = -x_1\}
\end{aligned}
$$

Therefore the problem in equation 24 is equivalent to the following standard form convex optimisation problem (for which the primal optimal cost is clearly 0, achieved when $x_1 = x_2 = 0$):

$$
\begin{aligned}
\underset{x \in \mathbb{R}^2}{\text{Minimise}} \quad & x_1^2 + x_2^2 \\
\text{Subject to} \quad & f_1(x) = x_1 \leq 0 \\
& h_1(x) = x_1 + x_2 = 0
\end{aligned}
$$

2.2 Hyperbolic Constraints And Second-Order Cones

A second-order cone problem (SOCP) is an optimisation problem that has the following form (see [1], equation 4.36, page 156):

$$
\begin{aligned}
\underset{x \in \mathcal{X}}{\text{Minimise}} \quad & f^T x \\
\text{Subject to} \quad & \|A_i x + b_i\|_2 \leq c_i^T x + d_i \quad i \in \{1, \dots, m\} \\
& Fx = g
\end{aligned}
\tag{25}
$$

The following equivalence can be used to express several different types of problems as SOCPs, which holds for any $x \in \mathbb{R}^n$ and $y, z \in \mathbb{R}$ which satisfy $y > 0$ and $z > 0$:

$$
x^T x \leq yz \quad \Leftrightarrow \quad \left\| \begin{bmatrix} 2x \\ y - z \end{bmatrix} \right\|_2 \leq y + z \tag{26}
$$

This equivalence can be proved as follows:

$$
\begin{aligned}
\left\|\begin{bmatrix} 2x \\ y-z \end{bmatrix}\right\|_2 &= \sqrt{\begin{bmatrix} 2x \\ y-z \end{bmatrix}^T \begin{bmatrix} 2x \\ y-z \end{bmatrix}} \\
&= \sqrt{(2x)^T(2x) + (y-z)^2} \\
&= \sqrt{4x^Tx + y^2 - 2yz + z^2}
\end{aligned}
$$

$$
\begin{aligned}
& & 0 \leq \left\|\begin{bmatrix} 2x \\ y-z \end{bmatrix}\right\|_2 &\leq y+z \\
&\Leftrightarrow & 0 \leq \sqrt{4x^Tx + y^2 - 2yz + z^2} &\leq y+z \\
&\Leftrightarrow & 4x^Tx + y^2 - 2yz + z^2 &\leq y^2 + 2yz + z^2 \\
&\Leftrightarrow & 4x^Tx &\leq 4yz \\
&\Leftrightarrow & x^Tx &\leq yz
\end{aligned}
$$

This equivalence can be used for example to show that the following optimisation problem, which can be interpreted as maximising a harmonic mean, is equivalent to a SOCP (a_i^T refers to the ith row of A):

$$
\begin{aligned}
\underset{x}{\text{Maximise}} \quad & \left(\sum_{i=1}^{m}\left[\frac{1}{a_i^Tx - b_i}\right]\right)^{-1} \\
\text{Subject to} \quad & Ax > b
\end{aligned}
\tag{27}
$$

This equivalence can be demonstrated as follows ($\cong$ is used to denote equivalence between optimisation problems, $\mathbf{0}$ in bold-face is used to denote an appropriately sized matrix or vector in which every element is equal to 0, $\mathbf{1}$ in bold-face is used to denote an appropriately sized vector in which every element is equal to 1, and e_i is used to denote the ith basis vector in which the ith element is equal to 1 and all other elements are equal to 0):

$$\begin{aligned}
& \underset{x}{\text{Maximise}} && \left(\sum_{i=1}^{m}\left[\frac{1}{a_i^T x - b_i}\right]\right)^{-1} \\
& \text{Subject to} && Ax > b \\[2ex]
\cong \quad & \underset{x}{\text{Minimise}} && \sum_{i=1}^{m}\left[\frac{1}{a_i^T x - b_i}\right] \\
& \text{Subject to} && Ax > b \\[2ex]
\cong \quad & \underset{x,y}{\text{Minimise}} && \mathbf{1}^T y \\
& \text{Subject to} && Ax > b \\
& && y_i(a_i^T x - b_i) = 1 \quad (\forall i) \\[2ex]
\cong \quad & \underset{x,y}{\text{Minimise}} && \mathbf{1}^T y \\
& \text{Subject to} && y \geq 0 \\
& && y_i(a_i^T x - b_i) \geq 1 \quad (\forall i) \\[2ex]
\cong \quad & \underset{x,y}{\text{Minimise}} && \mathbf{1}^T y \\
& \text{Subject to} && \|0\|_2 \leq y_i \quad (\forall i) \\
& && \left\|\begin{bmatrix} 2 \\ y_i - a_i^T x + b_i \end{bmatrix}\right\|_2 \leq y_i + a_i^T x - b_i \quad (\forall i) \\[2ex]
\cong \quad & \underset{x,y}{\text{Minimise}} && \begin{bmatrix} \mathbf{0} \\ \mathbf{1} \end{bmatrix}^T \begin{bmatrix} x \\ y \end{bmatrix} \\
& \text{Subject to} && \left\|\begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix}\begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \end{bmatrix}\right\|_2 \leq \begin{bmatrix} \mathbf{0} \\ e_i \end{bmatrix}^T \begin{bmatrix} x \\ y \end{bmatrix} + 0 \quad (\forall i) \\
& && \left\|\begin{bmatrix} \mathbf{0} & \mathbf{0} \\ -a_i^T & e_i^T \end{bmatrix}\begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} 2 \\ b_i \end{bmatrix}\right\|_2 \leq \begin{bmatrix} a_i \\ e_i \end{bmatrix}^T \begin{bmatrix} x \\ y \end{bmatrix} - b_i \quad (\forall i)
\end{aligned}$$

Which at last is in the form of a SOCP defined in equation 25.

The equivalence in equation 26 can also be used to show that a similar problem which can be interpreted as maximising a geometric mean can be reformulated as a SOCP (assuming m is a power of 2):

$$\begin{aligned}
& \underset{x}{\text{Maximise}} && \left(\prod_{i=1}^{m}\left[a_i^T x - b_i\right]\right)^{1/m} \\
& \text{Subject to} && Ax > b
\end{aligned} \tag{28}$$

The trick is to recursively define variables $t_{j,i}$ for $j \in \{1, \ldots, \log_2(m)\}$ as follows:

$$\begin{aligned}
t_{0,i} &= a_i^T x - b_i \\
t_{j+1,i} &= \sqrt{(t_{j,2i})(t_{j,2i+1})} \\
\Rightarrow \quad t_{j+1,i}^2 &\leq (t_{j,2i})(t_{j,2i+1}) \\
\Leftrightarrow \quad \left\|\begin{bmatrix} 2t_{j+1,i} \\ t_{j,2i} - t_{j,2i+1} \end{bmatrix}\right\|_2 &\leq t_{j,2i} + t_{j,2i+1}
\end{aligned}$$

The optimisation problem in equation 28 can therefore be expressed as an equivalent SOCP as follows:

$$
\begin{array}{lll}
\underset{x}{\text{Maximise}} & t_{\log_2(m),1} & \\
\text{Subject to} & t_{0,i} = a_i^T x - b_i & (\forall i) \\
& t_{j,i} \geq 0 & (\forall i)(\forall j \in \{0, \ldots, \log_2(m)\}) \\
& \left\| \begin{bmatrix} 2t_{j+1,i} \\ t_{j,2i} - t_{j,2i+1} \end{bmatrix} \right\|_2 \leq t_{j,2i} + t_{j,2i+1} & (\forall i)(\forall j \in \{0, \ldots, \log_2(m-1)\})
\end{array}
\tag{29}
$$

2.3 Support Functions

The support function S_C of a set C is defined as follows:

$$
S_C(y) = \sup_{x \in C} \left[y^T x \right] \tag{30}
$$

The support function S_C is convex, regardless of whether C is convex or not, which is proved below (the proof is similar to the proof that the Lagrangian dual function is concave, provided in appendix B). To start the proof, it is useful to define a variable $x^* \in \bar{C}$ (where $\bar{C}$ denotes the closure of the set C) which is chosen to satisfy the following equation, given $\lambda \in [0, 1]$, y, and z:

$$
\begin{aligned}
\sup_{x \in C} \left[\lambda y^T x + (1-\lambda) z^T x \right] &= \lambda y^T x^* + (1-\lambda) z^T x^* \\
&\Rightarrow \begin{cases} \sup_{x \in C} \left[y^T x \right] \geq y^T x^* \\ \sup_{x \in C} \left[z^T x \right] \geq z^T x^* \end{cases} \\
\Rightarrow \lambda \sup_{x \in C} \left[y^T x \right] + (1-\lambda) \sup_{x \in C} \left[z^T x \right] &\geq \lambda y^T x^* + (1-\lambda) z^T x^* \\
&= \sup_{x \in C} \left[\lambda y^T x + (1-\lambda) z^T x \right] \\
\Rightarrow (\forall y, z)(\forall \lambda \in [0,1]) \quad \lambda S_C(y) + (1-\lambda) S_C(z) &\geq S_C(\lambda y + (1-\lambda) z)
\end{aligned}
$$

It can also be prooved that $S_C = S_{\mathrm{conv}(C)}$, where $\mathrm{conv}(C)$ is the convex hull of C, defined below:

$$
\mathrm{conv}(C) = \{x : (\exists \lambda \in [0,1])(\exists a, b \in C) \quad x = \lambda a + (1-\lambda) b\} \tag{31}
$$

To prove that $S_C(y) = S_{\mathrm{conv}(C)}(y)$, it is useful to consider a point $x \in \mathrm{conv}(C)$, which by definition can be expressed as a convex combination of points $a, b \in C$:

$$
\begin{aligned}
(\forall x \in \mathrm{conv}(C))(\exists \lambda \in [0,1], a, b \in C) \quad x &= \lambda a + (1-\lambda) b \\
\Rightarrow (\forall y) \quad y^T x &= \lambda y^T a + (1-\lambda) y^T b \\
&\leq \lambda \sup_{a \in C} \left[y^T a \right] + (1-\lambda) \sup_{b \in C} \left[y^T b \right] \\
&= \lambda S_C(y) + (1-\lambda) S_C(y) \\
&= S_C(y) \\
\Rightarrow (\forall y) \quad \sup_{x \in \mathrm{conv}(C)} \left[y^T x \right] &\leq S_C(y) \\
\Rightarrow (\forall y) \quad S_{\mathrm{conv}(C)}(y) &\leq S_C(y)
\end{aligned}
$$

Therefore, for any y, $S_{\mathrm{conv}(C)}(y)$ is always a lower bound on $S_C(y)$. Because $C \subseteq \mathrm{conv}(C)$, from the definition of the support function in equation 30, we must also have that for any y, $S_{\mathrm{conv}(C)}(y) \geq S_C(y)$, by inclusion. Therefore it follows that for any y, $S_{\mathrm{conv}(C)}(y) = S_C(y)$, and therefore $S_{\mathrm{conv}(C)} = S_C$.

2.4 Largest-L Norm Of A Vector

For any vector $x \in \mathbb{R}^n$, the notation $x_{[i]}$ is used to denote the element of x with the ith largest magnitude, which implies the following ordering of the elements of x:

$$
|x_{[1]}| \geq |x_{[2]}| \geq \ldots \geq |x_{[n]}| \geq 0
$$

The largest-L norm of x is defined as follows:

$$\|x\|_{[L]} = \sum_{i=1}^{L} \left[\left| x_{[i]} \right| \right] \tag{32}$$

It can be proved that $f(x) = \|x\|_{[L]}$ is a convex function. First, it is useful to define the permutation ρ_x as the permutation which maps each integer $i \in \{1, \ldots, n\}$ to the ith largest element of x (with ties broken arbitrarily):

$$\begin{aligned}
\rho_x : \{1, \ldots, n\} &\to \{1, \ldots, n\} \\
x_{\rho_x(i)} &= x_{[i]} \\
\Rightarrow &\begin{cases} |x_{\rho_x(1)}| \geq |x_{\rho_x(2)}| \geq \ldots \geq |x_{\rho_x(n)}| \geq 0 \\ \|x\|_{[L]} = \sum_{i=1}^{L} \left[\left| x_{\rho_x(i)} \right| \right] \end{cases}
\end{aligned}$$

For any permutation $\rho : \{1, \ldots, n\} \to \{1, \ldots, n\}$ besides ρ_x, the following inequality must hold:

$$\sum_{i=1}^{L} \left[\left| x_{\rho_x(i)} \right| \right] \geq \sum_{i=1}^{L} \left[\left| x_{\rho(i)} \right| \right]$$

To prove this inequality, assume that the inequality were violated, then it must be the case that $\rho(i)$ for $i \in \{1, \ldots, L\}$ selects the L largest-in-magnitude elements of x, but ρ_x does not, which contradicts the definition of ρ_x.

Similarly, for any vector $y \in \mathbb{R}^n$ and any scalar $\lambda \in [0, 1]$, it is useful to define the permutations ρ_y and $\rho_{xy\lambda}$ on the integers $i \in \{1, \ldots, n\}$ as follows:

$$\begin{aligned}
x_{\rho_y(i)} &= y_{[i]} \\
x_{\rho_{xy\lambda}(i)} &= (\lambda x + (1 - \lambda) y)_{[i]}
\end{aligned}$$

Now it is straightforward to prove that $f(x) = \|x\|_{[L]}$ is a convex function, given any $x, y \in \mathbb{R}^n$ and any $\lambda \in [0, 1]$:

$$\begin{aligned}
\|\lambda x + (1 - \lambda) y\|_{[L]} &= \sum_{i=1}^{L} \left[\left| (\lambda x + (1 - \lambda) y)_{[i]} \right| \right] \\
&= \sum_{i=1}^{L} \left[\left| (\lambda x + (1 - \lambda) y)_{\rho_{xy\lambda}(i)} \right| \right] \\
&\leq \lambda \sum_{i=1}^{L} \left[\left| x_{\rho_{xy\lambda}(i)} \right| \right] + (1 - \lambda) \sum_{i=1}^{L} \left[\left| y_{\rho_{xy\lambda}(i)} \right| \right] \\
&\leq \lambda \sum_{i=1}^{L} \left[\left| x_{\rho_x(i)} \right| \right] + (1 - \lambda) \sum_{i=1}^{L} \left[\left| y_{\rho_y(i)} \right| \right] \\
&= \lambda \sum_{i=1}^{L} \left[\left| x_{[i]} \right| \right] + (1 - \lambda) \sum_{i=1}^{L} \left[\left| y_{[i]} \right| \right] \\
&= \lambda \|x\|_{[L]} + (1 - \lambda) \|y\|_{[L]}
\end{aligned}$$

This concludes the proof that $f(x) = \|x\|_{[L]}$ is convex.

The computation of $\|x\|_{[L]}$ for $x \in \mathbb{R}^N$ can be formulated as an integer programming problem as follows, where $(x_{\text{abs}})_i = |x_i|$:

$$\begin{aligned}
\underset{s \in \mathbb{Z}^N}{\text{Maximise}} \quad & x_{\text{abs}}^T s \\
\text{Subject to} \quad & \mathbf{1}^T s = L \\
& s_i \in \{0, 1\} \quad (\forall i \in \{1, \ldots, n\})
\end{aligned}$$

The computation of $\|x\|_{[L]}$ for $x \in \mathbb{R}^N$ can also be formulated as a linear program as follows:

$$\begin{aligned}
\underset{s \in \mathbb{R}^N}{\text{Maximise}} \quad & x_{\text{abs}}^T s \\
\text{Subject to} \quad & \mathbf{1}^T s = L \\
& 0 \leq s_i \leq 1 \quad (\forall i \in \{1, \ldots, n\})
\end{aligned}$$

The former approach can be interpreted as the support function of x_{abs} over the set $S_{\text{bin}} = \{s \in \{0,1\}^N : \mathbf{1}^T s = 1\}$, and the latter approach can be interpreted as the support function of x_{abs} over the set $S_{\text{cont}} = \{s \in [0,1]^N : \mathbf{1}^T s = 1\}$. The set S_{cont} is the convex hull of the set S_{bin}, therefore the results of section 2.3 imply that the two results should be equivalent.

3 Duality

3.1 Projection Onto The L1 Ball

A point $a \in \mathbb{R}^N$ can be projected onto the unit ball in the ℓ_1 norm by solving the following quadratic program:

$$\begin{aligned} \underset{x}{\text{Minimise}} \quad & \frac{1}{2}\|x-a\|_2^2 \\ \text{Subject to} \quad & \|x\|_1 \le c \end{aligned}$$

This is equivalent to the following quadratic program with slack variables t introduced:

$$\begin{aligned} \underset{x,t}{\text{Minimise}} \quad & \frac{1}{2}(x-a)^T(x-a) \\ \text{Subject to} \quad & x \le t \\ & x \ge -t \\ & \mathbf{1}^T t \le c \end{aligned}$$

The Lagrangian and Lagrangian dual functions can be defined for this problem as follows:

$$\begin{aligned} \mathcal{L}(x,t,\lambda) &= \frac{1}{2}(x-a)^T(x-a) + \lambda_1^T(x-t) + \lambda_2^T(-x-t) + \lambda_3(\mathbf{1}^T t - c) \\ &= \frac{1}{2}x^T x + (-a+\lambda_1-\lambda_2)^T x + \frac{1}{2}a^T a + (-\lambda_1-\lambda_2+\lambda_3\mathbf{1})^T t - \lambda_3 c \\ \frac{\partial \mathcal{L}}{\partial x} &= x + (-a+\lambda_1-\lambda_2) \\ \frac{\partial \mathcal{L}}{\partial x} = 0 \quad &\Rightarrow \quad x = a - \lambda_1 + \lambda_2 \\ &\Rightarrow \quad \mathcal{L}(x,t,\lambda) = -\frac{1}{2}(a-\lambda_1+\lambda_2)^T(a-\lambda_1+\lambda_2) + \frac{1}{2}a^T a + (-\lambda_1-\lambda_2+\lambda_3\mathbf{1})^T t - \lambda_3 c \\ g(\lambda) &= \inf_{x,t}\left[\mathcal{L}(x,t,\lambda)\right] \\ &= \begin{cases} -\frac{1}{2}(a-\lambda_1+\lambda_2)^T(a-\lambda_1+\lambda_2) + \frac{1}{2}a^T a - \lambda_3 c & -\lambda_1-\lambda_2+\lambda_3\mathbf{1} = 0 \\ -\infty & \text{Otherwise} \end{cases} \end{aligned}$$

The dual problem in this case is equivalent to maximising the dual function $g(\lambda)$ with respect to $\lambda \ge 0$, which can be expressed as the following quadratic program:

$$\begin{aligned} \underset{\lambda_1,\lambda_2,\lambda_3}{\text{Maximise}} \quad & -\frac{1}{2}(a-\lambda_1+\lambda_2)^T(a-\lambda_1+\lambda_2) + \frac{1}{2}a^T a - \lambda_3 c \\ \text{Subject to} \quad & -\lambda_1-\lambda_2+\lambda_3\mathbf{1} = 0 \\ & \lambda_1 \ge 0 \\ & \lambda_2 \ge 0 \\ & \lambda_3 \ge 0 \end{aligned}$$

The dual problem could be solved efficiently by using an interior point method, for example. From the solution to the dual problem, the optimal value for x can be found easily as the value that minimises the Lagrangian function, where λ_1 and λ_2 represent the Lagrange multipliers for the inequality constraints $x \le t$ and $x \ge -t$ respectively:

$$x = a - \lambda_1 + \lambda_2$$

3.2 SVM Duality

A standard support vector machine problem can be expressed as follows:

$$\min_{a,b}\left[\sum_i \left[\max\left[\{0, 1 - s_i(a^T v_i + b)\}\right]\right] + k\|a\|_2^2\right]$$

This problem can be expressed as a quadratic program by introducing the slack variables t, matrix V, and diagonal matrix S as follows:

$$\begin{aligned}
V_{ij} =& (v_i)_j \\
S_{ij} =& \begin{cases} s_i & i = j \\ 0 & i \neq j \end{cases} \\
\Rightarrow \quad \underset{a,b,t}{\text{Maximise}} \quad & \mathbf{1}^T t + k a^T a \\
& t \geq 0 \\
& t \geq \mathbf{1} - S(Va + b\mathbf{1})
\end{aligned}$$

The Lagrangian and Lagrangian dual functions can be defined for this problem as follows:

$$\begin{aligned}
\mathcal{L}(a,b,t,\lambda) &= \mathbf{1}^T t + k a^T a - \lambda_1^T t + \lambda_2^T(\mathbf{1} - S(Va + b\mathbf{1}) - t) \\
&= (\mathbf{1} - \lambda_1 - \lambda_2)^T t + k a^T a - \lambda_2^T S V a - \lambda_2^T S \mathbf{1} b + \lambda_2^T \mathbf{1} \\
\frac{\partial \mathcal{L}}{\partial a} &= 2ka - V^T S \lambda_2 \\
\frac{\partial \mathcal{L}}{\partial a} = 0 \quad &\Rightarrow \quad a = \frac{1}{2k} V^T S \lambda_2 \\
&\Rightarrow \quad \mathcal{L}(a,b,t,\lambda) = (\mathbf{1} - \lambda_1 - \lambda_2)^T t - \frac{1}{4k}\lambda_2^T S V V^T S \lambda_2 - \lambda_2^T S \mathbf{1} b + \lambda_2^T \mathbf{1} \\
g(\lambda) &= \inf_{a,b,t} \left[\mathcal{L}(a,b,t,\lambda)\right] \\
&= \begin{cases} -\frac{1}{4k}\lambda_2^T S V V^T S \lambda_2 + \lambda_2^T \mathbf{1} & \mathbf{1} - \lambda_1 - \lambda_2 = 0 \quad \text{and} \quad \lambda_2^T S \mathbf{1} = 0 \\ -\infty & \text{Otherwise} \end{cases}
\end{aligned}$$

The dual problem in this case is equivalent to maximising the dual function $g(\lambda)$ with respect to $\lambda \geq 0$, which can be expressed as the following quadratic program:

$$\begin{aligned}
\underset{\lambda_1,\lambda_2,\lambda_3}{\text{Maximise}} \quad & -\frac{1}{4k}\lambda_2^T S V V^T S \lambda_2 + \lambda_2^T \mathbf{1} \\
\text{Subject to} \quad & \mathbf{1} - \lambda_1 - \lambda_2 = 0 \\
& \lambda_2^T S \mathbf{1} = 0 \\
& \lambda_1 \geq 0 \\
& \lambda_2 \geq 0 \\
& \lambda_3 \geq 0
\end{aligned}$$

3.3 Adjustable Optimization

In an uncertain optimisation problem, a decision x is made, after which a vector w is observed, which initially is only known to lie in the set $\mathcal{W} = \{w : Fw \leq \mathbf{1}\}$, and the decision is then updated to $x + f(w)$. To simplify the problem, it can be assumed that $f(w)$ is linear, which is to say that $f(w) = Mw$ for some matrix M, and the problem becomes choosing both x and M to minimise some worst-case cost, depending on the (initially unknown, but later observed) value of w. This scenario can be expressed as the following optimisation problem:

$$\begin{aligned}
\underset{M,x}{\text{Minimise}} \quad & \max_{w \in \mathcal{W}} \Big[c^T(x + Mw)\Big] \\
\text{Subject to} \quad & A(x + Mw) \leq b + Bw \quad (\forall w \in \mathcal{W})
\end{aligned}$$

An upper bound on the worst-case cost can be found using the Lagrangian dual function of the following problem:

$$\begin{aligned}
\underset{w}{\text{Maximise}} \quad & c^T(x + Mw) \\
\text{Subject to} \quad & Fw \leq \mathbf{1}
\end{aligned}$$

The Lagrangian and Langrangian dual functions for this problem can be defined as follows:

$$\begin{aligned}
\mathcal{L}(w,\lambda) &= c^T(x + Mw) + \lambda^T(Fw - \mathbf{1}) \\
g(\lambda) &= \sup_w \Big[\mathcal{L}(w,\lambda)\Big] \\
&= \begin{cases} c^T x - \lambda^T \mathbf{1} & M^T c + F^T \lambda = 0 \\ \infty & \text{Otherwise} \end{cases}
\end{aligned}$$

The Lagrangian dual function is an upper bound on the worst-case cost for all values of $\lambda \geq 0$, therefore the tightest upper bound on the worst-case cost can be expressed as the solution to the following optimisation problem:

$$\begin{aligned} \underset{\lambda}{\text{Minimise}} \quad & c^T x - \lambda^T \mathbf{1} \\ \text{Subject to} \quad & M^T c + F^T \lambda = 0 \\ & \lambda \geq 0 \end{aligned}$$

The original optimisation problem can now be expressed as minimising the tightest upper bound on the worst-case cost:

$$\begin{aligned} \underset{M,x,\lambda}{\text{Minimise}} \quad & c^T x - \lambda^T \mathbf{1} \\ \text{Subject to} \quad & (AM - B)w \leq b - Ax \quad (\forall w \in \mathcal{W}) \\ & M^T c + F^T \lambda = 0 \\ & \lambda \geq 0 \end{aligned}$$

The problem now requires finding bounds on each element of the vector $(AM - B)w$ for all $w \in \mathcal{W}$, which itself depends on the decision variable M, and the previously found constraints on M. The greatest value that the ith element of this vector can take can itself be expressed as the solution to another optimisation problem, where a_i^T and b_i^T refer to the ith rows of the matrices A and B, respectively:

$$\begin{aligned} \underset{w}{\text{Maximise}} \quad & (a_i^T M - b_i^T)w \\ \text{Subject to} \quad & Fw \leq \mathbf{1} \\ & M^T c + F^T \lambda = 0 \end{aligned}$$

The Lagrangian and Langrangian dual functions for this problem can be defined as follows:

$$\begin{aligned} \mathcal{L}_i(w, \mu_i, \nu_i) &= (a_i^T M - b_i^T)w + \mu_i^T (Fw - \mathbf{1}) + \nu_i^T (M^T c + F^T \lambda) \\ &= (M^T a_i - b_i + F^T \mu_i)^T w - \mu_i^T \mathbf{1} + \nu_i^T (M^T c + F^T \lambda) \\ g_i(\mu_i, \nu_i) &= \sup_w \Big[\mathcal{L}_i(w, \mu_i, \nu_i) \Big] \\ &= \begin{cases} -\mu_i^T \mathbf{1} + \nu_i^T (M^T c + F^T \lambda) & M^T a_i - b_i + F^T \mu_i = 0 \\ \infty & \text{Otherwise} \end{cases} \end{aligned}$$

The Lagrangian dual function is an upper bound on $(a_i^T M - b_i^T)w$ for all values of $\mu_i \geq 0$ and ν_i, therefore the tightest upper bound on $(a_i^T M - b_i^T)w$ can be expressed as the solution to the following optimisation problem:

$$\begin{aligned} \underset{\mu_i,\nu_i}{\text{Minimise}} \quad & -\mu_i^T \mathbf{1} + \nu_i^T (M^T c + F^T \lambda) \\ \text{Subject to} \quad & M^T a_i - b_i + F^T \mu_i = 0 \\ & \mu_i \geq 0 \end{aligned}$$

At last, the solution to the original optimisation problem can be expressed as follows, wherein $[b]_i$ refers to the ith element of the vector b, to be distinguished from b_i^T which refers to the ith row of B:

$$\begin{aligned} \underset{M,x,\lambda,\{\mu_i\}_i,\{\nu_i\}_i}{\text{Minimise}} \quad & c^T x - \lambda^T \mathbf{1} \\ \text{Subject to} \quad & M^T c + F^T \lambda = 0 \\ & M^T a_i - b_i + F^T \mu_i = 0 && (\forall i) \\ & -\mu_i^T \mathbf{1} + \nu_i^T (M^T c + F^T \lambda) \leq [b]_i - a_i^T x && (\forall i) \\ & \mu_i \geq 0 && (\forall i) \\ & \lambda \geq 0 \end{aligned}$$

4 Algorithms

The following is an example of a linear program in standard form, where $x \in \mathbb{R}^n$ and $A \in \mathbb{R}^{m \times n}$:

$$\begin{aligned} \underset{x}{\text{Minimise}} \quad & c^T x \\ \text{Subject to} \quad & Ax = b \\ & x \geq 0 \end{aligned}$$

The Lagrangian and Langrangian dual functions for this problem can be defined as follows:

$$\begin{aligned}
\mathcal{L}(x,\lambda,\nu) &= c^T x - \lambda^T x + \nu^T (Ax-b) \\
&= (c - \lambda + A^T\nu)^T x - \nu^T b \\
g(\lambda,\nu) &= \inf_x \Big[\mathcal{L}(x,\lambda,\nu)\Big] \\
&= \begin{cases} -\nu^T b & c-\lambda+A^T\nu = 0 \\ -\infty & \text{Otherwise} \end{cases}
\end{aligned}$$

The dual problem can be expressed as follows:

$$\begin{aligned}
\underset{\lambda,\nu}{\text{Maximise}} \quad & -\nu^T b \\
\text{Subject to} \quad & c - \lambda + A^T\nu = 0 \\
& \lambda \geq 0
\end{aligned}$$

The Karush-Kuhn-Tucker (KKT) conditions for this problem are as follows:

$$\begin{cases}
Ax = b & \multirow{2}{*}{\text{Primal feasibility}} \\
x \geq 0 & \\
\lambda \geq 0 & \text{Dual feasibility} \\
x_i\lambda_i = 0 \quad (\forall i) & \text{Complementary slackness} \\
c - \lambda + A^T\nu = 0 & \text{Lagrangian stationarity}
\end{cases}$$

The KKT conditions are not linear in the variables x, λ and ν, in particular because the complementary slackness conditions are bilinear. However, the complementary slackness conditions can be replaced with the condition that the primal optimal cost is equal to the dual optimal cost (which is equivalent to the condition of strong duality), because the two could not be equal if the complementary slackness conditions were not satisfied (this can be seen from the proof in section 1 that the dual optimal cost is a lower bound of the primal optimal cost, in which equality can only hold if the complementary slackness conditions are satisfied). Therefore, the following is a set of sufficient conditions for the solution to this linear program:

$$\begin{cases}
Ax = b & \multirow{2}{*}{\text{Primal feasibility}} \\
x \geq 0 & \\
\lambda \geq 0 & \text{Dual feasibility} \\
c^T x = -\nu^T b & \text{Strong duality} \\
c - \lambda + A^T\nu = 0 & \text{Lagrangian stationarity}
\end{cases}$$

One method for solving the linear program is the alternating projection method, which consists of alternately projecting $z = (x,\nu,\lambda)$ onto the sets $\mathcal{A}$ and $\mathcal{C}$ respectively, where $\mathcal{A}$ and $\mathcal{C}$ are defined as follows:

$$\begin{aligned}
\mathcal{A} &= \left\{ z : F \begin{pmatrix} x \\ \nu \\ \lambda \end{pmatrix} = g \right\} \\
\text{where} \quad F &= \begin{pmatrix} c^T & b^T & 0 \\ A & 0 & 0 \\ 0 & A^T & -I \end{pmatrix} \\
g &= \begin{pmatrix} 0 \\ b \\ -c \end{pmatrix} \\
\mathcal{C} &= \Big\{ z : x \geq 0 \quad \text{and} \quad \lambda \geq 0 \Big\}
\end{aligned}$$

Projection onto the set $\mathcal{C}$ can easily be performed simply by clipping negative values of x and λ to 0. Projection onto the set $\mathcal{A}$ can be performed by solving the following optimisation problem:

$$\begin{aligned}
\underset{y}{\text{Minimise}} \quad & \|y - z\|_2^2 \\
\text{Subject to} \quad & Fy = g
\end{aligned}$$

The solution is straightforward to derive using Lagrange multipliers, and is equal to the standard least-squares solution:

$$y = z + F^T \left(FF^T\right)^{-1} (g - Fz)$$

In the interest of efficient implementation, because the matrix FF^T does not change from one iteration to the next, its LU/Cholesky decomposition can be stored and reused in every iteration.

References

[1] Stephen Boyd and Lieven Vandenberghe. *Convex optimization*. Cambridge university press, 2004.

A Motivation For The Lagrangian Function

Consider the following optimisation problem, with decision variable $x \in \mathbb{R}^n$, objective function $f_0(x)$, a single equality constraint $h(x)$, and no inequality constraints:

$$\begin{aligned} \underset{x \in \mathcal{X}}{\text{Minimise}} \quad & f_0(x) \\ \text{Subject to} \quad & h(x) = 0 \end{aligned}$$

The objective and constraint functions have the following first-order Taylor expansions:

$$\begin{aligned} f_0(x + \delta) &= f_0(x) + \delta^T \frac{\partial f_0}{\partial x} + \ldots \\ h(x + \delta) &= h(x) + \delta^T \frac{\partial h}{\partial x} + \ldots \end{aligned}$$

Assuming a feasible point $\tilde{x}$ has been found (which implies that $h(\tilde{x}) = 0$), a feasible direction $\tilde{\delta}$ in which to move must satisfy $\tilde{\delta}^T \frac{\partial h}{\partial x} = 0$, because moving in this direction does not (to first order) change the value of $h(\tilde{x})$, and therefore does not violate the constraint $h(\tilde{x} + \tilde{\delta}) = 0$. If there is a feasible direction $\tilde{\delta}$ for which $\tilde{\delta}^T \frac{\partial f_0}{\partial x} < 0$, then moving in direction $\tilde{\delta}$ will improve the value of the objective function. Therefore, at a local optimum of the feasible set, at which there is no feasible direction that improves the value of the objective function, it must be the case that any feasible direction $\tilde{\delta}$ (which must satisfy $\tilde{\delta}^T \frac{\partial h}{\partial x} = 0$) must also satisfy $\tilde{\delta}^T \frac{\partial f_0}{\partial x} = 0$ (if this was not true, we could feasibly move in direction $\pm\tilde{\delta}$ and improve the objective function, in which case this point would not be a local optimum of the feasible set). If all vectors $\tilde{\delta}$ which are satisfy $\tilde{\delta}^T \frac{\partial h}{\partial x} = 0$ also satisfy $\tilde{\delta}^T \frac{\partial f_0}{\partial x} = 0$, then it must be the case that $\frac{\partial f_0}{\partial x}$ and $\frac{\partial h}{\partial x}$ are parallel to each other, for some constant of proportionality λ:

$$\begin{aligned} & \frac{\partial f_0}{\partial x} \propto \frac{\partial h}{\partial x} \\ \Rightarrow (\exists \lambda) \quad & \frac{\partial f_0}{\partial x} = -\lambda \frac{\partial h}{\partial x} \\ \Rightarrow \quad & \frac{\partial f_0}{\partial x} + \lambda \frac{\partial h}{\partial x} = 0 \\ \Rightarrow \quad & \frac{\partial}{\partial x}\Big[f_0(x) + \lambda h(x)\Big] = 0 \\ \Rightarrow \quad & \frac{\partial}{\partial x}\Big[\mathcal{L}(x, \lambda)\Big] = 0 \\ \text{where} \quad & \mathcal{L}(x, \lambda) = f_0(x) + \lambda h(x) \end{aligned}$$

If the function $\mathcal{L}(x, \lambda)$ is also stationary with respect to λ, then this implies that the constraint $h(x)$ is satisfied, and therefore we have found a local optimum of the feasible set.

In the case of multiple equality constraint functions, $h_i(x) = 0$ for $i \in \{1, \ldots, p\}$, a feasible direction $\tilde{\delta}$ is one for which $\tilde{\delta}^T \frac{\partial h_i}{\partial x} = 0$ for all $i \in \{1, \ldots, p\}$. Starting from a feasible point $\tilde{x}$, if the gradient of the objective function $\frac{\partial f_0}{\partial x}$ has a component which is orthogonal to the linear subspace spanned by the gradients of the constraint functions $\frac{\partial h_i}{\partial x}$ for $i \in \{1, \ldots, p\}$, we can feasibly move in $\pm$ this direction, while still satisfying all constraints, and improving the value of the objective function. Therefore at a local optimum of the feasible set, at which there is no feasible direction in which to move that improves the value of the objective function, it must be the case that the gradient of the objective function has no component which is orthogonal to the linear subspace spanned by the gradients of the constraint functions (otherwise we could move in $\pm$ this direction while satisfying the constraints and improving the objective function), which implies that the gradient the gradient of the objective function is in the linear subspace spanned by the gradients of the constraint functions, and therefore can be expressed as a linear combination of the gradients of the constraint functions:

$$
\begin{aligned}
(\exists \lambda \in \mathbb{R}^p) \quad \frac{\partial f_0}{\partial x} &= -\sum_{i=1}^{p}\left[\lambda_i \frac{\partial h_i}{\partial x}\right] \\
\Rightarrow \quad \frac{\partial}{\partial x}\left[f_0(x) + \sum_{i=1}^{p}\left[\lambda_i h_i(x)\right]\right] &= 0 \\
\Rightarrow \quad \frac{\partial}{\partial x}\left[\mathcal{L}(x,\lambda)\right] &= 0 \\
\text{where} \quad \mathcal{L}(x,\lambda) &= f_0(x) + \sum_{i=1}^{p}\left[\lambda_i h_i(x)\right]
\end{aligned}
$$

If the function $\mathcal{L}(x,\lambda)$ is also stationary with respect to λ, then this implies that the all constraints $h_i(x) = 0$ for $i \in \{1,\ldots,p\}$ are satisfied, and therefore we have found a local optimum of the feasible set.

B Proof That The Lagrangian Dual Function Is Concave

We start by proving equation 7, for which it is useful to define the variable $x^* \in \bar{\mathcal{X}}$ (where $\bar{\mathcal{X}}$ denotes the closure of the set $\mathcal{X}$) which is chosen to satisfy the following equation, given $\alpha \in [0,1]$, $\lambda^{(1)}$, $\lambda^{(2)}$, and ν:

$$
\begin{aligned}
\inf_{x\in\mathcal{X}}\left[\mathcal{L}(x, \alpha\lambda^{(1)} + (1-\alpha)\lambda^{(2)}, \nu)\right] &= \mathcal{L}(x^*, \alpha\lambda^{(1)} + (1-\alpha)\lambda^{(2)}, \nu) && (33) \\
&\Rightarrow \begin{cases} \inf_{x\in\mathcal{X}}\left[\mathcal{L}(x, \lambda^{(1)}, \nu)\right] \leq \mathcal{L}(x^*, \lambda^{(1)}, \nu) \\ \inf_{x\in\mathcal{X}}\left[\mathcal{L}(x, \lambda^{(2)}, \nu)\right] \leq \mathcal{L}(x^*, \lambda^{(2)}, \nu) \end{cases} && (34) \\
\Rightarrow \alpha \inf_{x\in\mathcal{X}}\left[\mathcal{L}(x, \lambda^{(1)}, \nu)\right] + (1-\alpha)\inf_{x\in\mathcal{X}}\left[\mathcal{L}(x, \lambda^{(2)}, \nu)\right] &\leq \alpha\mathcal{L}(x^*, \lambda^{(1)}, \nu) + (1-\alpha)\mathcal{L}(x^*, \lambda^{(2)}, \nu) && (35) \\
&= \mathcal{L}(x^*, \alpha\lambda^{(1)} + (1-\alpha)\lambda^{(2)}, \nu) && (36) \\
&= \inf_{x\in\mathcal{X}}\left[\mathcal{L}(x, \alpha\lambda^{(1)} + (1-\alpha)\lambda^{(2)}, \nu)\right] && (37) \\
\Rightarrow (\forall\lambda^{(1)}, \lambda^{(2)} \in \mathbb{R}^m)(\forall\alpha \in [0,1]) \quad \alpha g(\lambda^{(1)}, \nu) + (1-\alpha)g(\lambda^{(2)}, \nu) &\leq g(\alpha\lambda^{(1)} + (1-\alpha)\lambda^{(2)}, \nu) && (38)
\end{aligned}
$$

Where (36) follows from the definition of the Lagrangian function $\mathcal{L}$ in equation 4 and the fact that $\alpha + (1-\alpha) = 1$, and (38) follows from the definition of the Lagrangian dual function g in equation 5. The proof that the dual Lagrangian function is concave with respect to ν (equation 8) follows using similar reasoning.