Lecture 11

We continue the study of algorithms for linear programming. Before doing so, we introduce a **standard form** for a linear programming problem, namely

$$\text{minimize} \quad \langle \boldsymbol{c}, \boldsymbol{x}\rangle \qquad \text{subject to} \quad \boldsymbol{A}\boldsymbol{x} = \boldsymbol{b},\ \boldsymbol{x} \geq \boldsymbol{0}. \tag{P}$$

for a matrix $\boldsymbol{A} \in \mathbb{R}^{m\times n}$, $\boldsymbol{b} \in \mathbb{R}^m$ and $\boldsymbol{c} \in \mathbb{R}^n$. Any linear programming problem can be brought into this standard for. Start, for example, with the problem

$$\text{maximize} \quad \langle \boldsymbol{c}, \boldsymbol{x}\rangle \qquad \text{subject to} \quad \boldsymbol{A}\boldsymbol{x} \leq \boldsymbol{b}, \tag{1.1}$$

in the form that we are used to. We first add a **slack variable** $\boldsymbol{s} \in \mathbb{R}^m$, so that we can reformulate the problem into an equalent one of the form

$$\text{maximize} \quad \langle \boldsymbol{c}, \boldsymbol{x}\rangle \qquad \text{subject to} \quad \boldsymbol{A}\boldsymbol{x} + \boldsymbol{s} = \boldsymbol{b}, \boldsymbol{s} \geq \boldsymbol{0}. \tag{1.2}$$

Define the big matrix $\boldsymbol{A}' \in \mathbb{R}^{m\times(n+n+m)}$ by

$$\boldsymbol{A}' := \begin{pmatrix} \boldsymbol{A} & -\boldsymbol{A} & \boldsymbol{I} \end{pmatrix},$$

consider the big vector $\boldsymbol{x}' = (\boldsymbol{x}^+, \boldsymbol{x}^-, \boldsymbol{s})^\top$, where $\boldsymbol{x}^+$ is the *positive part*, with entries x_i if $x_i \geq 0$ and 0 else, and $\boldsymbol{x}^-$ is the *negative part*, with entries $-x_i$ if $x_i < 0$ and 0 else (so that $\boldsymbol{x} = \boldsymbol{x}^+ - \boldsymbol{x}^-$), and set $\boldsymbol{c}' = (\boldsymbol{c}, -\boldsymbol{c}, \boldsymbol{0})^\top \in \mathbb{R}^{2n+m}$.

Then $\boldsymbol{x}$ is a solution of (1.2) if and only if $\boldsymbol{x}'$ is a solution of the problem

$$\text{minimize} \quad \langle \boldsymbol{c}', \boldsymbol{x}'\rangle \qquad \text{subject to} \quad \boldsymbol{A}'\boldsymbol{x}' = \boldsymbol{b},\ \boldsymbol{x}' \geq \boldsymbol{0},$$

which is in standard form (P). Note that since (P) has the form of a dual problem as derived in Lecture 9 and 10, its dual, in turn, has the form of a primal problem,

$$\text{maximize} \quad \langle \boldsymbol{b}, \boldsymbol{y}\rangle \qquad \text{subject to} \quad \boldsymbol{A}^\top\boldsymbol{y} + \boldsymbol{s} = \boldsymbol{c},\ \boldsymbol{s} \geq \boldsymbol{0}. \tag{D}$$

For everything that follows, we assume $m \leq n$, as otherwise (D) is unbounded and (P) empty.

1.1 An optimality condition

We now work with a primal dual system *in standard form*,

$$\text{minimize} \quad \langle \boldsymbol{c}, \boldsymbol{x} \rangle \quad \text{subject to} \quad \boldsymbol{A}\boldsymbol{x} = \boldsymbol{b},\ \boldsymbol{x} \geq \boldsymbol{0}, \tag{P}$$

and

$$\text{maximize} \quad \langle \boldsymbol{b}, \boldsymbol{y} \rangle \quad \text{subject to} \quad \boldsymbol{A}^\top \boldsymbol{y} + \boldsymbol{s} = \boldsymbol{c},\ \boldsymbol{s} \geq \boldsymbol{0}. \tag{D}$$

A useful consequence of the duality theorem, which states that a primal optimal solution is also dual optimal, is a characterisation of solution tuples $(\boldsymbol{x}^*, \boldsymbol{y}^*, \boldsymbol{s}^*)$. Note that for such an optimal pair,

$$\langle \boldsymbol{c}, \boldsymbol{x}^* \rangle = \langle \boldsymbol{y}^*, \boldsymbol{b} \rangle = \langle \boldsymbol{y}^*, \boldsymbol{A}\boldsymbol{x}^* \rangle = \langle \boldsymbol{A}^\top \boldsymbol{y}^*, \boldsymbol{x}^* \rangle,$$

where the first equality holds because of the duality theorem. By subtracting the last from the first expression above, we conclude

$$\langle \boldsymbol{x}^*, \boldsymbol{c} - \boldsymbol{A}^\top \boldsymbol{y}^* \rangle = \langle \boldsymbol{x}^*, \boldsymbol{s}^* \rangle = 0. \tag{PD}$$

Since $\boldsymbol{x}^* \geq \boldsymbol{0}$ and $\boldsymbol{s}^* \geq \boldsymbol{0}$, each summand in (PD) is zero. This means that the individual components of $\boldsymbol{s}^*$ and $\boldsymbol{x}^*$ satisfy

$$x_i^* \cdot s_i^* = 0,\ 1 \leq i \leq m.$$

Summarising, we have the following *optimality conditions* for linear programming: if vectors $(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{s})$ are primal/dual optimal solutions to a linear programming problem, then

$$\begin{aligned} \boldsymbol{A}\boldsymbol{x} + \boldsymbol{s} - \boldsymbol{b} &= \boldsymbol{0} \\ \boldsymbol{A}^\top \boldsymbol{y} - \boldsymbol{c} &= \boldsymbol{0} \\ y_i s_i &= 0,\ 1 \leq i \leq m \\ \boldsymbol{y} &\geq \boldsymbol{0} \\ \boldsymbol{s} &\geq \boldsymbol{0}. \end{aligned} \tag{1.1}$$

We simplify this expression a bit further. Define the diagonal matrices

$$\boldsymbol{X} = \begin{pmatrix} x_1 & 0 & \cdots & 0 \\ 0 & x_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & x_n \end{pmatrix}, \quad \boldsymbol{S} = \begin{pmatrix} s_1 & 0 & \cdots & 0 \\ 0 & s_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & s_n \end{pmatrix}$$

and the vector $\boldsymbol{e} = (1, 1, \dots, 1)^\top$. Then the condition $x_i s_i = 0$ for $1 \leq i \leq n$ can be written concisely as $\boldsymbol{X}\boldsymbol{S}\boldsymbol{e} = \boldsymbol{0}$, and the whole system as

$$\begin{aligned} \boldsymbol{A}^\top \boldsymbol{y} + \boldsymbol{s} - \boldsymbol{c} &= \boldsymbol{0} \\ \boldsymbol{A}\boldsymbol{x} - \boldsymbol{b} &= \boldsymbol{0} \\ \boldsymbol{X}\boldsymbol{S}\boldsymbol{e} &= \boldsymbol{0} \\ \boldsymbol{x} &\geq \boldsymbol{0} \\ \boldsymbol{s} &\geq \boldsymbol{0}, \end{aligned} \tag{1.2}$$

Just as the optimality condition $\nabla f(\boldsymbol{x}) = \boldsymbol{0}$ serves as the basis for algorithms for unconstraint optimization, the optimality conditions for linear programming form the basis of the simplex method and of interior point methods. In the simplex method, the conditions are used to verify whether a candidate vertex is an optimal point, while primal/dual interior point methods view (1.2) as a multivariate function in $(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{s})$ for which a root satisfying the inequality constraints is sought using Newton's method or similar algorithms.

1.2 Newton's method for solving equations

Recall that we have seen Newton's method in two forms: as a minimization algorithm, and as a method for finding roots of a non-linear equation. In the latter, in the one-dimensional setting we want to solve an equation

$$f(x) = 0$$

and proceed by starting with an initial guess x_0, and then computing successively

$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}.$$

If f is the derivative of another function g, then this method is used to find a local minimizer of g. For a function $F = (f_1, \dots, f_n)^\top \colon \mathbb{R}^n \to \mathbb{R}^n$ we can use the same method, only that dividing by the derivative is replaced with multiplying with the inverse of the Jacobian matrix,

$$\boldsymbol{J}F(\boldsymbol{x}) = \begin{pmatrix} \frac{\partial f_1}{\partial x_1} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_n}{\partial x_1} & \cdots & \frac{\partial f_n}{\partial x_n} \end{pmatrix}.$$

Newton's method then starts with an initial guess $\boldsymbol{x}_0$, and proceeds by

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k - \alpha_k \boldsymbol{J}F(\boldsymbol{x}_k)^{-1} F(\boldsymbol{x}_k).$$

In practise, one does not compute the inverse, but solves a system of equations to get the update,

$$\boldsymbol{J}F(\boldsymbol{x}_k)\Delta\boldsymbol{x}_k = F(\boldsymbol{x}_k), \quad \boldsymbol{x}_{k+1} = \boldsymbol{x}_k - \alpha_k \Delta\boldsymbol{x}_k. \tag{1.1}$$

Note that we allow for adjusting the step length in Newton's method. This can be rather convenient.

We can apply the multivariate Newton's method to the equalities in the optimality condition (1.2), by considering the function $F \colon \mathbb{R}^{2n+m} \to \mathbb{R}^{2n+m}$ defined by

$$F(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{s}) = \begin{pmatrix} \boldsymbol{A}^\top \boldsymbol{y} + \boldsymbol{s} - \boldsymbol{c} \\ \boldsymbol{A}\boldsymbol{x} - \boldsymbol{b} \\ \boldsymbol{X}\boldsymbol{S}\boldsymbol{e} \end{pmatrix}$$

We are then looking for a root $(\boldsymbol{x}^*, \boldsymbol{y}^*, \boldsymbol{s}^*)$ of this function that in addition satisfies the inequality constraints in (1.2). Before we get into the non-trivial issue enrusing non-negativity, we first have a look at what the *update step* (1.1) looks like. Computing the Jacobian for F, the updates are computed by solving

$$\begin{pmatrix} \boldsymbol{0} & \boldsymbol{A}^\top & \boldsymbol{I} \\ \boldsymbol{A} & \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{S} & \boldsymbol{0} & \boldsymbol{X} \end{pmatrix} \begin{pmatrix} \Delta \boldsymbol{x} \\ \Delta \boldsymbol{y} \\ \Delta \boldsymbol{s} \end{pmatrix} = \begin{pmatrix} \boldsymbol{A}^\top \boldsymbol{y} + \boldsymbol{s} - \boldsymbol{c} \\ \boldsymbol{A}\boldsymbol{x} - \boldsymbol{b} \\ \boldsymbol{X}\boldsymbol{S}\boldsymbol{e} \end{pmatrix}$$

Interior point method change the condition the condition $\boldsymbol{X}\boldsymbol{S}\boldsymbol{e} = \boldsymbol{0}$ to $\boldsymbol{X}\boldsymbol{S}\boldsymbol{e} = \tau\boldsymbol{e}$ for some $\tau > 0$, thus ensuring that we get non-negative solutions. Solving this problem for values $\tau \to 0$ then gives an approximation of the true solution. We discuss this in more detail in the next lecture.