Non-Linear Programming
Lecture notes

Emmanuel Rachelson

2016

These lecture notes cover the basic (and not-so-basic) notions used to solve problems of non-linear, differentiable optimization. They are (or at least should be) no substitute for a full textbook or a live class. The intent is to provide a concise, yet self-sufficient reminder of the main results in the field. Consequently, these notes are meant to be used in two situations:

1. In the preparation of a Nonlinear Optimization class, to gain knowledge of the main topics before class.
2. As a memento after an Optimization class, to quickly recall the main results.

Readers interested in a full-length textbook are encouraged to consult the excellent "Nonlinear Programming" by Prof. D. P. Bertsekas for instance.

1 Non-Linear Programming problems

Let f, g_i ($i \in [1,q]$) et h_j ($j \in [1,p]$) be functions mapping $\mathbb{R}^n$ to $\mathbb{R}$. These functions are assumed to be continuous and twice differentiable.
Solving the Nonlinear Optimization (or Nonlinear Programming) problem associated to these functions consists in finding $\hat{x} \in \mathbb{R}^n$ such that:

$$f(x) = \min_{x \in \Omega} f(x)$$

$$\Omega = \left\{ x \in \mathbb{R}^n / \begin{array}{c} \forall i \in [1,q], g_i(x) = 0, \\ \forall j \in [1,p], h_j(x) \leq 0 \end{array} \right\}$$

This problem corresponds to finding the minimum of function f in the domain defined by the g_i and h_j constraints. Function f is called the *objective function*. We write g the function mapping $\mathbb{R}^n$ to $\mathbb{R}^q$, linking x to the vector of all $g_i(x)$. Similarly, $h(x)$ is the vector of all $h_j(x)$.
We shall refer to this problem as (P_Ω).

2 Differentiability and convexity

This part recalls the essential notions of functional analysis required for optimization. It can be skipped if you are at ease with the notions of gradient, Hessian and convexity.
Consider a function $f : \mathbb{R}^n \to \mathbb{R}$. The gradient of f in x is:

$$\nabla f(x) = \begin{pmatrix} \frac{\partial f}{\partial x_1}(x) \\ \vdots \\ \frac{\partial f}{\partial x_n}(x) \end{pmatrix}$$

The notation $\frac{\partial f}{\partial x}(x) = [\nabla f(x)]^T$ is often used.

The slope of f, in x, in direction v is $f'(x)(v) = v^T \nabla f(x)$.
Let $f : \mathbb{R}^n \to \mathbb{R}$ be twice differentiable, and $x \in \mathbb{R}^n$. The Hessian of f in x is the matrix:

$$\nabla^2 f(x) = \frac{\partial^2 f}{\partial x^2}(x) = \begin{bmatrix} \frac{\partial^2 f}{\partial x_1^2}(x) & \dots & \frac{\partial^2 f}{\partial x_1 \partial x_n}(x) \\ \vdots & \ddots & \vdots \\ \frac{\partial^2 f}{\partial x_n \partial x_1}(x) & \dots & \frac{\partial^2 f}{\partial x_n^2}(x) \end{bmatrix}$$

The Hessian is symmetric, so $\frac{\partial^2 f}{\partial x_i \partial x_j}(x) = \frac{\partial^2 f}{\partial x_j \partial x_i}(x)$.
The slope of $f'(x)(v)$, when x moves infinitesimally in direction w, is $f''(x)(v)(w) = v^T \left[\nabla^2 f(x)\right] w$. So, in particular, $\frac{\partial^2 f}{\partial x_i \partial x_j}(x)$ measures how much the slope of f along direction x_i changes along direction x_j.
One can approximate a function locally with Taylor's second order expansion:

$$\begin{aligned} f(x+v) &= f(x) + f'(x)(v) + \frac{1}{2} f''(x)(v)(v) + o(\|v\|^2) \\ &= f(x) + v^T \nabla f(x) + v^T \left[\nabla^2 f(x)\right] v + o(\|v\|^2) \end{aligned}$$

We move on to define convexity. A set $X \subset \mathbb{R}^n$ is convex iff $\forall (x,y) \in X^2,\ \forall \lambda \in [0,1],\ \lambda x + (1-\lambda) y \in X$.
Let X be a convex subset of $\mathbb{R}^n$. A function $f : X \to \mathbb{R}$ is called convex if

$$\forall (x,y) \in X^2, \forall \lambda \in [0;1],\ f\left(\lambda x + (1-\lambda) y\right) \leq \lambda f(x) + (1-\lambda) f(y)$$

The function f is called strictly convex if the above inequality is strict. It is called concave is $-f$ is convex.
One can use the gradient to characterize the convexity of a function. Let X be a convex subset of $\mathbb{R}^n$ and $f : \mathbb{R}^n \to \mathbb{R}$ be differentiable over X:

$$f \text{ is convex} \Leftrightarrow \forall (x,y) \in X^2,\ f(y) \geq f(x) + (y-x)^T \nabla f(x)$$

The function f is strictly convex if that inequality is strict for $x \neq y$.
Finally, the Hessian of f in x fully describes the shape of f, and thus its convexity. Let X be a convex subset of $\mathbb{R}^n$ and $f : \mathbb{R}^n \to \mathbb{R}$ be twice differentiable over X:

1. f convex over $X \Leftrightarrow\ \forall x \in X,\ \nabla^2 f(x) \geq 0$ ($\nabla^2 f(x)$ is positive semi-definite)
2. f strictly convex over $X \Leftrightarrow\ \forall x \in X,\ \nabla^2 f(x) > 0$ ($\nabla^2 f(x)$ is positive definite)
3. f strictly convex over an open neighbourhood of $x \in X \Leftrightarrow\ \nabla^2 f(x) > 0$

The above proposition indicates that if all eigen-values of the Hessian in x are positive (resp. strictly positive), then the function is convex (resp. strictly convex). The two first items in the proposition are global versions (the function is globally convex), the third one is a local version.

3 Non-linear optimization theorems

3.1 Unconstrained optimization

In the absence of constraints, minimas of f can be found via the following first and second order conditions.
Theorem (First order condition)**.** *x is a local solution to $(P_{\mathbb{R}^n}) \Rightarrow \frac{\partial f}{\partial x}(x) = 0$*

Theorem (Second order condition)**.** $\left.\begin{array}{l} \frac{\partial f}{\partial x}(x) = 0 \\ \frac{\partial^2 f}{\partial x^2}(x) \textit{ positive definite} \end{array}\right\} \Leftrightarrow x \textit{ is a local solution of } (P_{\mathbb{R}^n})$

Remarks:

- The first order condition is a necessary condition.
- The second order condition is both necessary and sufficient.
- If f is strictly convex over $\mathbb{R}^n$, then the first order condition is both necessary and sufficient.

3.2 Optimizing over an open subset

Theorem. *Let Ω be an open subset of $\mathbb{R}^n$. x is a solution of $(P_\Omega) \Leftrightarrow \left\{ \begin{array}{c} x \in \Omega \\ x \text{ is a solution of } (P_{\mathbb{R}^n}) \end{array} \right.$*

Explanation: this might seem trivial but it needs to be stated, if all solutions x of $(P_{\mathbb{R}^n})$ are outside of the open set Ω (that is, if they are on its border or further), then (P_Ω) has no solutions at all. Conversely: if (P_Ω) has a solution, then this solution is among the solutions of $(P_{\mathbb{R}^n})$ and it is inside Ω.
In the end, this is why we only tackle constraints of the form $h_j(x) \leq 0$ and never consider $h_j(x) < 0$. If we had to consider some $h_j(x) < 0$ constraints, we would take the relaxed version $h_j(x) \leq 0$ into account, and then check that the solution found verifies the strict formulation.

3.3 Lagrange theorem

In this part we only consider equality constraints $g_i(x) = 0$ that define the admissible domain Ω. We define the *Lagrangian* function:

$$\forall (x, \lambda) \in \mathbb{R}^n \times \mathbb{R}^q, L(x, \lambda) = f(x) + \sum_{i=1}^{q} \lambda_i g_i(x) = f(x) + \lambda^T g(x)$$

We also say v is a feasible variation (or an admissible direction) iff:

$$\exists \phi \in C^1(\mathbb{R}^+, \mathbb{R}^n) / \left\{ \begin{array}{c} \phi(0) = x \\ \phi'(0) = v \\ \phi(t) \in \Omega \text{ for } t \text{ small enough} \end{array} \right.$$

The set of all feasible variations in x in noted $\mathcal{D}_x(\Omega)$.
Remarks:

- An admissible direction is a direction “in which one can move, from x, without leaving the admissible domain”.
- For equality constraints, one has $\mathcal{D}_x(\Omega) = \ker \left[\frac{\partial g}{\partial x}(x) \right] = \left\{ v \in \mathbb{R}^n / v^T \nabla g(x) = 0 \right\}$.

In the theorem below, we note $\nabla_x \mathcal{L}(x) = \left[\frac{\partial L}{\partial x}(x, \lambda) \right]^T$ and $\nabla_\lambda L(x) = \left[\frac{\partial \mathcal{L}}{\partial \lambda}(x, \lambda) \right]^T$ to avoid any ambiguities and keep the notation simple.

Theorem (Lagrange theorem)**.** *If, at a point $x^* \in \mathbb{R}^n$, the constraints gradients $\{\nabla g_i(x^*), i \in [1, q]\}$ are linearly independent:*

1. *First order, necessary condition:*

$$x^* \text{ is a local minimum of } (P_\Omega) \Rightarrow \exists \lambda^* \in \mathbb{R}^q \ / \ \left\{ \begin{array}{c} \nabla_x L(x^*, \lambda^*) = 0 \\ g(x^*) = 0 \end{array} \right\}$$

2. *Second order, sufficient condition:*

$$\exists \lambda^* \in \mathbb{R}^q \ / \ \left\{ \begin{array}{c} \nabla_x L(x^*, \lambda^*) = 0 \\ g(x^*) = 0 \\ \forall v \in \mathcal{D}_{x^*}(\Omega) \setminus \{0\}, v^T \frac{\partial^2 L}{\partial x^2}(x^*, \lambda^*) v > 0 \end{array} \right\} \Leftrightarrow x^* \text{ is a local minimum of } (P_\Omega)$$

Remarks:

- The λ_i coefficients are called *Lagrange multipliers.*

- The condition that $\{\nabla g_i(x^*), i \in [1,q]\}$ be linearly independent is called *constraint regularity.* If the constraints satisfy this condition in x, then x is called *regular.* If x is a local minimum and x is regular, then the Lagrange multipliers are unique.

- A point x is regular iff $\nabla g(x)$ has rank q.

- The condition $g(x^*) = 0$ just states that the minimum found must be in the admissible domain and can be rephrased $\nabla_\lambda L(x^*, \lambda^*) = 0$ since $\nabla_\lambda L(x, \lambda) = g(x)$

- The direct implication ($\Rightarrow$) in the second order conditions actually does not require x^* to be regular (only the reverse implication requires it for the unicity of λ).

- In order to find a minimum to f under constraints, one first tries to solve the system of equations provided by the first order condition ($n+q$ equations, $n+q$ variables). If a minimum of f under constraints exists, then it must verify these equations. Thus, this system of equations provides *candidates* for optimality. Then, for each of these candidates, one verifies if the Lagrangian function is indeed convex in every feasible direction $v^T \frac{\partial^2 L}{\partial x^2}(x^*, \lambda^*)v > 0$. If so, then the candidate is a local minimum. To find the global minimum, one simply compares the values of f in local minima.

3.4 Karush-Kuhn-Tucker theorem

Let us now consider the general case of equality and inequality constraints. The Lagrangian now is written:

$$\forall (x, \lambda, \mu) \in \mathbb{R}^n \times \mathbb{R}^q \times \mathbb{R}^p, \ L(x, \lambda, \mu) = f(x) + \sum_{i=1}^{q} \lambda_i g_i(x) + \sum_{j=0}^{p} \mu_j h_j(x) = f(x) + \lambda^T g(x) + \mu^T h(x)$$

If, at a point $x \in \mathbb{R}^n$, a given constraint verifies $h_j(x) = 0$, then this constraint is called *active* or *saturated.* Let $\mathcal{A}(x)$ the set of the active constraints indices at x.
The set of admissible directions in x now becomes:

$$\mathcal{D}_x(\Omega) = \left\{ v \in \mathbb{R}^n / v^T \nabla g(x) = 0 \text{ and } v^T \nabla h_j(x) \leq 0 \text{ for } j \in \mathcal{A}(x) \right\}$$

At a point x, the set of active constraints gradients is: $\{\nabla g_i(x), i \in [1,q]\} \cup \{\nabla h_j(x), j \in \mathcal{A}(x)\}$

Theorem (Karush-Kuhn-Tucker theorem). *If, at a point $x^* \in \mathbb{R}^n$, the active constraints gradients are linearly indedepent:*

1. *First order, necessary condition:*

$$x^* \text{ is a local minimum of } (P_\Omega) \Rightarrow \exists\, (\lambda^*, \mu^*) \in \mathbb{R}^q \times \mathbb{R}^{+p} \ / \ \left\{ \begin{array}{c} \nabla_x L(x^*, \lambda^*) = 0 \\ g(x^*) = 0 \\ \forall j \in [1,p], \mu_j^* h_j(x^*) = 0 \end{array} \right\}$$

2. *Second order, sufficient condition: if there exist $(\lambda^*, \mu^*) \in \mathbb{R}^q \times \mathbb{R}^{+p}$ such that*

$$\left\{ \begin{array}{c} \nabla_x L(x^*, \lambda^*, \mu^*) = 0 \\ \nabla_\lambda L(x^*, \lambda^*, \mu^*) = 0 \\ \forall j \in [1,p], \mu_j^* h_j(x^*) = 0 \\ \forall v \in \mathcal{D}_{x^*}(\Omega) \setminus \{0\}, v^T \nabla_x^2 L(x^*, \lambda^*, \mu^*) v > 0 \end{array} \right\} \Leftrightarrow x^* \text{ is a local minimum of } (P_\Omega)$$

Remarks:

- This formulation of KKT's theorem is a somewhat "weak" formulation which does not allow to deal with some degenerate cases. However, it is enough for the general understanding of the principles and, for most cases, it will be sufficient, so we shall keep it without further development.

- The condition imposing that $\{\nabla g_i(x), i \in [1,q]\} \cup \{\nabla h_j(x), j \in \mathcal{A}(x)\}$ be linearly independent is a particular case (a sufficient condition for) of a more general condition called *constraint qualification.*
- A particular case of constraint qualification: if $\{g_i, i \in [1,q]\} \cup \{h_j, j \in \mathcal{A}(\hat{x})\}$ are linear and different from each other, then the constraints are also qualified.
- Warning: the μ_j belong to $\mathbb{R}^+$.
- The λ_i and μ_j coefficients are called *Lagrange multipliers.* Sometimes the μ_j can be called KKT multipliers.
- In order to find a minimum to f under constraints, one first tries to solve the system of equations provided by the first order condition ($n+q+p$ equations, $n+q+p$ variables). If a minimum of f under constraints exists, then it must verify these equations. Thus, this system of equations provides *candidates* for optimality. Then, for each of these candidates, one verifies if the Lagrangian function is indeed convex in every feasible direction $v^T \frac{\partial^2 L}{\partial x^2}(x^*, \lambda^*) v > 0$. If so, then the candidate is a local minimum. To find the global minimum, one simply compares the values of f in local minima.

3.5 Duality

Let us consider only inequality constraints. We define the *primal problem* as the problem:

$$\inf_{x \in \mathbb{R}^n} \sup_{\mu \geq 0} L(x, \mu)$$

Similarly, we define the *dual problem* as:

$$\sup_{\mu \geq 0} \inf_{x \in \mathbb{R}^n} L(x, \mu)$$

One can then introduce:

Theorem (Duality). *Assume that f and $(h_j)_{j \in [1,p]}$ are convex. Then:*

$$\begin{aligned} x^* \textit{ is a solution of } (P_\Omega) &\Leftrightarrow \exists \mu^* \in \mathbb{R}^{+p} \ / \ (x^*, \mu^*) \textit{ is a solution of the primal problem} \\ &\Leftrightarrow \exists \mu^* \in \mathbb{R}^{+p} \ / \ (x^*, \mu^*) \textit{ is a solution to the dual problem} \end{aligned}$$

In other words:

x^* is a solution of (P_Ω) with the μ^* Lagrange multipliers $\Leftrightarrow$ (x^*, μ^*) is a saddle point of the Lagrangian function of (P_Ω).

4 In practice: analytical resolution of nonlinear programming problems

4.1 Using the KKT conditions

Let's start from the problem:

$$(P_\Omega) \ : \ \begin{cases} \min\limits_{x \in \mathbb{R}^n} f(x) \\ \forall i \in [1,q], g_i(x) = 0 \\ \forall j \in [1,p], h_j(x) \leq 0 \\ \forall k \in [1,l], m_k(x) < 0 \end{cases}$$

1. We first relax the m_k constraints and merge them with the h_j. This is because of the optimization over open sets theorem. We will need to check *[a posteriori]* that the found solution is indeed in the open set defined by the m_k constraints.

2. We verify that the constraints g_i and h_j are qualified (everywhere in the admissible domain).

3. We write the Lagrangian function.

4. We apply KKT theorem and find the derivatives of the Lagrangian function, yielding the $n+p+q$ equations (with $n+p+q$ unknowns):

$$\begin{gathered}\frac{\partial L}{\partial x}(x,\lambda,\mu)=0\\ \forall i\in[1,q], g_i(x)=0\\ \forall j\in[1,p], \mu_j h_j(x)=0\end{gathered}$$

   Solving this system of equations is arguably the most tedious step, since it requires to process all the different cases of "$\mu_j=0$ or $h_j(x)=0$" resulting from the $\mu_j h_j(x)=0$ equations. It is often relevant to start looking for a solution with all $\mu_j=0$ (corresponding to the absence of hypothesis on the constraints saturation) and to progressively saturate the constraints one after the other. However, this simple heuristic, although reasonable, has no guarantee to help find the best solution rapidly.
   It is always a good idea to order your calculations in an organized way, in order to be able to group similar cases together.
   This first order resolution yields a set of candidate points $(x^*,\lambda^*,\mu*)$. Among these points are the minima of our problem.

5. In every $(x^*,\lambda^*,\mu*)$ candidate point, we verify that $\forall v\in\mathcal{D}_{x^*}(\Omega)\setminus\{0\}, v^T\frac{\partial^2 L}{\partial x^2}(x^*,\lambda^*,\mu^*)v>0$. In general, we try to prove it for all $v\in\mathbb{R}^n$, then, if it appears infeasible, we specialize our attempts on vectors $v\in\mathcal{D}_{x^*}(\Omega)\setminus\{0\}$. Finally, if this fails too, we try to prove that the condition is not verified. Drawing figures often helps!

6. Then we check if the remaining $(x^*,\lambda^*,\mu*)$ candidates satisfy the m_k constraints.

7. The points $(x^*,\lambda^*,\mu*)$ found are then guaranteed to be local minima of f under all constraints. One can then search for the global minimum by comparing them together.

Remarks:

- The search for positive μ_j comes from the fact that we wrote the inequality constraints with the $\leq$ operator. If one changes the sign of these constraints, the sign of μ_j changes too. In order to avoid unintentional mistakes or confusions, it is advisable to always write the constraints in Nonlinear Programming problems in the same way, *i.e.* with "$=0$" equality constraints and "≤ 0" inequality constraints.

- Similarly, the search for positive (semi-)definite $\frac{\partial^2 L}{\partial x^2}$ comes from the fact that we search for a minimum. To avoid confusions or mistakes, it is advisable to always write optimization problems as minimization ones.

- Many exercices involve a reduced number of x variables. Some everyday-life problems too. In these cases, draw a figure! It really helps in getting an intuition of what to look for (especially when checking which candidates are minima or not). In other cases, the number of constraints is limited. Think about using the duality theorem and, again, draw a quick sketch. Unfortunately it does not always work (most of real-life non trivial problems have many variables) but it would be a shame not to do so when it is possible.

- Sometimes, some equality constraints can lead directly to the elimination (substitution) of some variables. Don't rush into calculation and take some time to consider the problem globally (get some physical sense about what is going on). Usually, taking some time to visualize Ω and the general shape of f is a good start.

4.2 Using the duality theorem

We start with the problem:

$$(P_\Omega)\ :\ \left\{ \begin{array}{c} \min\limits_{x\in\mathbb{R}^n} f(x) \\ \forall j \in [1,p], h_j(x) \leq 0 \end{array} \right.$$

1. Verify that f and h_j are convex (at least locally and restrict the admissible domain accordingly).
2. Write the Lagrangian function.
3. Solve the unconstrained optimization problem $\inf\limits_{x\in\mathbb{R}^n} L(x,\mu)$. For this purpose, write the derivatives of $L(x,\mu)$ with respect to x, and solve $\nabla_x L(x,\mu) = 0$ You obtain $x^*(\mu)$.
4. Write the dual function $\psi(\mu) = L(\hat{x}(\mu),\mu)$.
5. Solve the dual problem $\sup\limits_{\mu\geq 0} \psi(\mu)$ by writing $\nabla_\mu \psi = 0$ and distinguish the different cases according to the sign of μ.

5 Quadratic programming

Among Nonlinear programming problems, Quadratic problems are so common they deserve a short paragraph of their own. These problems are found in Optimal Control, Economy, Structural dimensioning, Operations Research, etc. A quadratic programming problem often corresponds to minimizing an energy-like quantity under design constraints describing the physical limits of the variables.
Consider $Q \in \mathbb{R}^{n\times n}$, b and $a_j \in \mathbb{R}^n$, k and $c_j \in \mathbb{R}$. Let f be the function $f(x) = \frac{1}{2}x^T Qx + b^T x + k$ et $h_j(x) = a_j^T x + c_j$.
The general form of a quadratic programming problem is then:ue.

$$\begin{gathered} \min_{x\in\mathbb{R}^n} \frac{1}{2} x^T Qx + b^T x + k \\ a_j^T x + c_j \leq 0 \end{gathered}$$

Their resolution is straightforward with KKT's theorem.

Remarks:

- All constraints are qualified (thus all points are regular) if the pairs (a_j, c_j) are all different from each other.
- The Lagrangian's gradient $\nabla_x L(x,\mu) = Qx + b + \sum\limits_{j=1}^{q} \mu_j a_j$.
- The Lagrangian's Hessian is constant and $\nabla^2 L(x,\mu) = Q$.
- The unconstrained solution is $-Q^{-1}b$ is Q is positive definite (so if f is convex, and Q invertible).

6 Algorithms

Coming shortly.

One can distinguish three categories of algorithms for optimizing differentiable functions, namely:

- Line search algorithm that find the minimum of a univariate function $f(t), t \in \mathbb{R}$.
- Unconstrained optimization algorithms that find the minimum of $f(x), x \in \mathbb{R}^n$ and often use line search methods as a building block.
- Constrained optimization algorithms, that use all of the above.

6.1 Line search methods

- Brent's method (parabolic interpolation)
- Golden section method
- Armijo rule
- Goldstein's rule
- Wolfe's rule

6.2 Unconstrained optimization

- Gradient descent
- Conjugate gradients
- Newton methods
- Quasi-Newton methods (DFP and BFGS)
- Nelder-Mead (simplex)
- COBYLA

6.3 Constrained optimization

- Conditional gradients method
- Gradient projection method
- Penalization method
- Uzawa's algorithm
- Sequential Quadratic Programming

7 Exercices

7.1 Differentiability and convexity

7.2 Unconstrained optimization

7.3 Lagrange theorem

7.4 Karush-Kuhn-Tucker theorem

7.5 Duality