

1 Genetic Algorithms and Convex Programming (Section 11.7)

This section will go over both generic algorithms and convex programming.

1.1 Genetic Algorithms

Inspired by observations in the biological sciences, **generic algorithms** can be used to solve combinatorial and discrete problems. Let's suppose we have a function $f(x)$. We don't know the minimum value of $f(x)$, only that its minimum value is greater than or equal to 0. In other words, $f(x) \geq 0$.

Suppose we have a population of $x^{(1)}, x^{(2)}, \dots$, which are solution estimates. We want to use $\frac{1}{F(x^{(i)})}$ as proxies for probabilities. Picking the solution estimates with high probabilities, we can recombine that solution estimates to form a new "generation" of solution estimates.

1.2 Convex Sets and Functions

We say that f is convex if it is defined on a convex set K and satisfies a condition. Before we go over the condition, let us define what a convex set is.

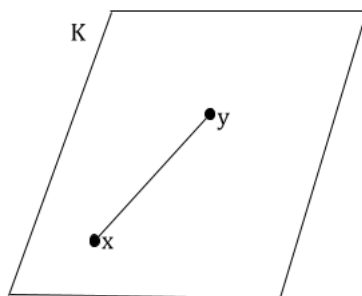
Definition 1.1: Convex Set

A **convex set** is a set K such that, for $x, y \in K$ and $0 \leq \theta \leq 1$,

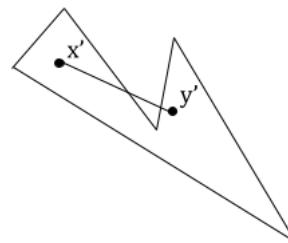
$$\theta x + (1 - \theta)y \in K.$$

In other words, for any two points in the set, the set contains the line between the two points.

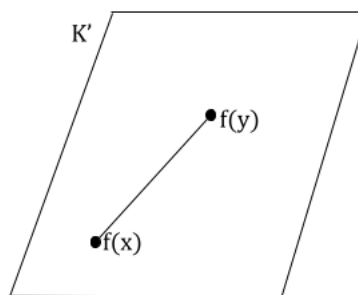
Convex Set



Not Convex Set



With this in mind, a convex function is analogous to a convex set; consider the visualization below:



But, we're considering the function values instead. So, if $f(x)$ and $f(y)$ form a line and that line remains within the set of function values, then that function is convex. More formally, for $x, y \in K'$ and $0 \leq \alpha \leq 1$, we have

$$\alpha f(x) + (1 - \alpha)f(y) \in K'.$$

1.3 Linear Inequalities & Consistent Systems

Fix $x \in \mathbb{R}^n$. Each linear inequality is defined by the constant $a_{ji}x_i - b_j \geq 0$. This, in turn, means that

$$\sum_{i=1}^n a_{ji}x_i - b_j \geq 0 \quad i \leq j \leq m.$$

This corresponds to the matrix inequality,

$$Ax - b \geq 0.$$

How do we know if the system is *consistent*? We say that a system is consistent if it has at least *one* solution.

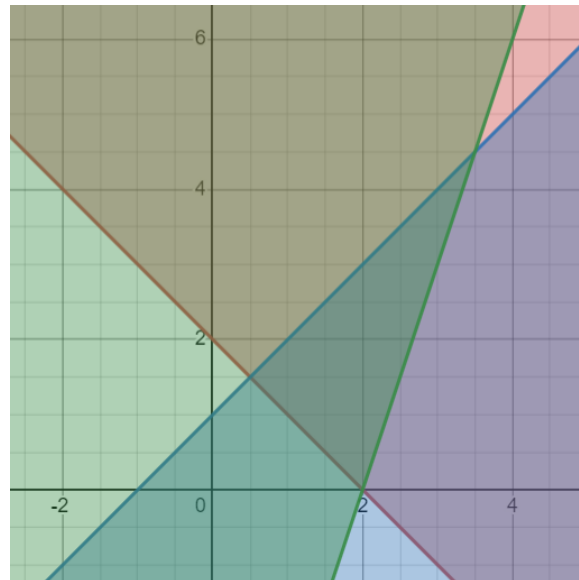
(Example.) Suppose $m = 3$ and $n = 2$, and suppose we have the system

$$\begin{bmatrix} 1 & 1 \\ 1 & -1 \\ -3 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} - \begin{bmatrix} 2 \\ -1 \\ -6 \end{bmatrix} \geq \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}.$$

The corresponding inequalities are:

$$\begin{aligned} x_1 + x_2 &\geq 2 \\ x_1 - x_2 &\geq -1 \\ -3x_1 + x_2 &\geq -6. \end{aligned}$$

In the two-dimensional case, we can plot the system and see what the corresponding regions are. By rewriting the system as appropriate, the resulting plots are shown below:



The feasible region is defined by the intersection of all three inequalities. In this case, the triangle where the three inequalities overlap above is our feasible region. This also defines our convex set: if we choose any two points in that region, and any linear combination, then the resulting line will remain in this region.

So, to summarize:

- The feasible region is defined by the intersection of all constraints (e.g., inequalities).
- If the feasible region is *empty*, then the system is inconsistent.

1.4 Convex Programming

In this area, a convex function can be approximated by

$$F(x) = \max_{1 \leq i \leq j} \left(\sum_{j=1}^n a_{ij}x_j - b_i \right).$$

The convex domain is approximated by the set

$$K = \{x : G(x) \leq 0\},$$

where

$$G(x) = \max_{k < i \leq m} \left(\sum_{j=1}^n a_{ij}x_j - b_i \right).$$

One way to condense this is to represent this problem in terms of a matrix A and a vector b . So, we can let $A \in \mathbb{R}^{m \times n}$ and vector $b \in \mathbb{R}^m$ describe the problem. Let's suppose that each set of n rows from A is linearly independent¹. Then, the residual value is defined by

$$r_i(x) = \sum_{j=1}^n a_{ij}x_j - b_i \quad 1 \leq i \leq m.$$

Moving to the idea of convex programming itself, we will consider an outline of an algorithm for how convex programming (optimization) works. Generally, we start with a set of indices $J \subset \{1, 2, 3, \dots, n\}$. Then, at each step,

1. J has $n + 1$ elements.
2. At least one element of J is in $\{1, 2, \dots, k\}$.
3. The origin is in the convex hull of $\{A_{\text{row } i} : i \in J\}$.

Also, at each step, we want to compute $x \in \mathbb{R}^n$ and $\lambda \in \mathbb{R}$ such that

$$\begin{aligned} r_i(x) &= \lambda & \text{if } i \in J & \quad (i \leq k) \\ r_i(x) &= 0 & \text{if } i \in J & \quad (i > k) \end{aligned}$$

This process gives us the solution of the linear system with $n + 1$ unknowns and $n + 1$ equations. To turn this into a rough algorithm, given x and λ , we want to compute $F(x)$ and $G(x)$.

- If $G(x) \leq 0$ and $F(x) = \lambda$, we can stop.
- If $G(x) > 0$, we can select an index $\lambda > k$ such that $r_\lambda(x) > 0$.
- If $G(x) \leq 0$ and $F(x) > \lambda$, then we can select an index $\alpha \leq k$ where $r_\alpha(x) > \lambda$.

Essentially, we want to replace α with an index $\beta \in J$ so that the new index set J' contains the origin in the convex hull.

¹This is known as the “Haar” condition.