Chapter 7. Concave Programming

Xiaoxiao Hu

March 30, 2020

Introduction

- In this chapter, we will combine the idea of convexity with a more conventional calculus approach.

- The result is that the Lagrange or Kuhn-Tucker conditions, in conjunction with convexity properties of the objective and constraint functions, are **sufficient** for optimality.

7.A. Concave Functions and Their Derivatives

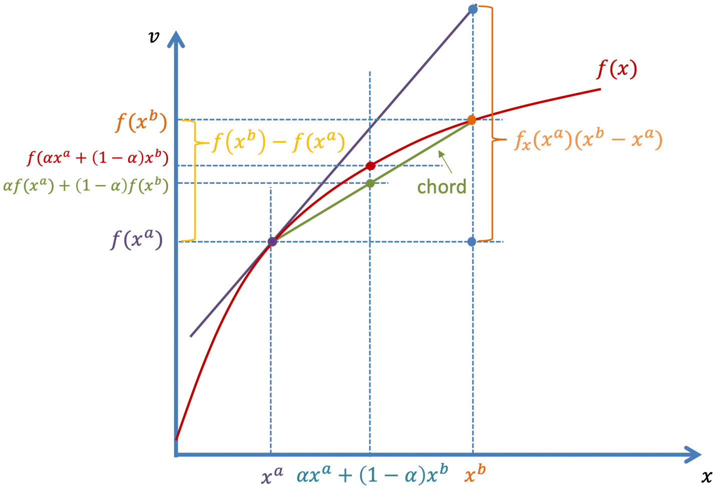
- The first step is to express the concavity (convexity) of functions in terms of their derivatives.

Definition 6.B.5 (Concave Function). *A function $f : \mathcal{S} \to \mathbb{R}$, defined on a convex set $\mathcal{S} \subset \mathbb{R}^N$, is concave if*

$$f(\alpha x^a + (1-\alpha)x^b) \geq \alpha f(x^a) + (1-\alpha) f(x^b), \tag{6.5}$$

for all x^a, $x^b \in \mathcal{S}$ and for all $\alpha \in [0, 1]$.

Concave Function



Concave Function

- To express the concavity of $f(x)$ in terms of its derivative, we now draw the tangent to $f(x)$ at x^a.
- The requirement of concavity says that the graph of the function should lie on or below the tangent.
- Or expressed differently,

$$f_x(x^a)(x^b - x^a) \geq f(x^b) - f(x^a),$$

where $f_x(x^a)$ is the slope of the tangent to $f(x)$ at x^a.

Concave Function

- Such an expression holds for higher dimensions.
- The result is summarized in Proposition 7.A.1 below.

Proposition 7.A.1 (Concave Function). *A differentiable function $f : \mathcal{S} \to \mathbb{R}$, defined on a convex set $\mathcal{S} \subset \mathbb{R}^N$, is concave if and only if*

$$f_x(x^a)(x^b - x^a) \geq f(x^b) - f(x^a), \tag{7.1}$$

for all $x^a, x^b \in \mathcal{S}$.

Convex Function

Similarly, for a differentiable convex function f, we have

$$f_x(x^a)(x^b - x^a) \leq f(x^b) - f(x^a). \tag{7.2}$$

7.B. Concave Programming

- A particularly important class of optimization problems has a concave objective function and convex constriant functions.

- The term *concave programming* is often used to describe the general problem of this kind.

Concave Programming

Consider the maximization problem

$$\max_x F(x)$$
$$\text{s.t. } G(x) \leq c,$$

where F is differentiable and concave, and each component constraint function G^i is differentiable and convex.

Concave Programming

We will interpret the problem using the terminology of the production problem:

$$\max_x \underbrace{F(x)}_{\text{revenue from outputs}}$$

$$\text{s.t. } \underbrace{G(x) \le c}_{\text{resource constraints}},$$

- x: the vector of outputs
- c: a fixed vector of input supplies
- $G(x)$: the vector of inputs needed to produce x
- $X(c)$: the optimum choice funcion
- $V(c)$: the maximum value function

Concave Programming

Claim 1. *$V(c)$ is a non-decreasing function.*

- x that was feasible for a given value of c remains feasible when any component of c increases, so the maximum value cannot decrease.

Concave Programming

Claim 2. *$V(c)$ is a concave.*

- To show concavity of $V(c)$, we need to show that for any two input supply vectors c and c' and any number $\alpha \in [0, 1]$, we have

$$V(\alpha c + (1-\alpha)c') \geq \alpha V(c) + (1-\alpha)V(c').$$

- That is, it should be possible to achieve revenue at least as high as $\alpha V(c) + (1-\alpha)V(c')$ when the input supply vector is $\alpha c + (1-\alpha)c'$.

Claim 2: $V(c)$ is a concave.

- Let $x^* = X(c)$ and $x^{*\prime} = X(c')$.
- Since the optimal choices must be feasible, we have

$$G(x^*) \leq c \quad \text{and} \quad G(x^{*\prime}) \leq c'. \tag{7.3}$$

- We will show that the output vector $\alpha x^* + (1 - \alpha x^*)'$ is feasible under the input supply vector $\alpha c + (1 - \alpha)c'$.
- And that it yields revenue at least as high as

$$\alpha V(c) + (1 - \alpha)V(c').$$

Claim 2: $V(c)$ is a concave.

(i) $\alpha x^* + (1-\alpha)x^{*\prime}$ is feasible since for each i, the convexity of G^i implies

$$G^i(\alpha x^* + (1-\alpha)x^{*\prime}) \underbrace{\leq}_{\text{convexity}} \alpha G^i(x^*) + (1-\alpha)G^i(x^{*\prime})$$

$$\underbrace{\leq}_{(7.3)} \alpha c_i + (1-\alpha)c_i^{\prime}.$$

Claim 2: $V(c)$ is a concave.

(ii) $\alpha x^* + (1-\alpha)x^{*\prime}$ yields revenue at least as high as $\alpha V(c) + (1-\alpha)V(c')$ since the concavity of F implies

$$F(\alpha x^* + (1-\alpha)x^{*\prime}) \underbrace{\geq}_{\text{concavity}} \alpha F(x^*) + (1-\alpha)F(x^{*\prime})$$

$$= \alpha V(c) + (1-\alpha)V(c'). \qquad (7.4)$$

Claim 2: $V(c)$ is a concave.

- Therefore, we have found a feasible output vector that generates the target revenue.
- The maximum revenue must be no smaller than the revenue generated from the feasible output vector:

$$V(\alpha c + (1-\alpha)c') \geq F(\alpha x^* + (1-\alpha)x^{*\prime}). \tag{7.5}$$

- With $F(\alpha x^* + (1-\alpha)x^{*\prime}) \geq \alpha V(c) + (1-\alpha)V(c')$, (7.4)

  we have $V(\alpha c + (1-\alpha)c') \geq \alpha V(c) + (1-\alpha)V(c')$. $\square$

Claim 2: $V(c)$ is a concave (Intuition)

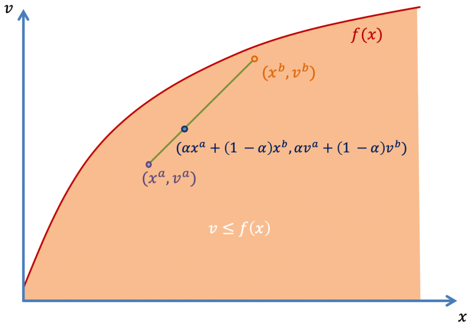
- The convexity of G rules out economies of scale or specialization in production, ensuring that a weighted average of outputs can be produced using the same weighted average of inputs.

- The concavity of F ensures that the resulting revenue is at least as high as the same weighted average of the separate revenues.

Concave Function

Recall the alternative interpretation of a concave function:

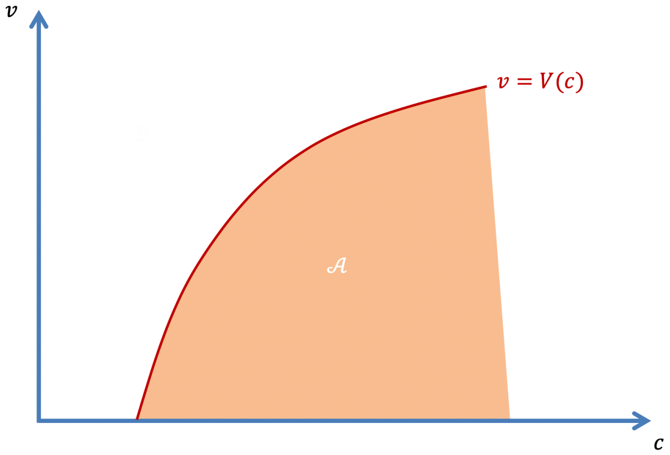
Claim. *f is a concave function if and only if $\mathcal{F} = \{(x, v) | v \leq f(x)\}$ is a convex set.*



Concave Function

- In our current context, as $V(c)$ is a concave function, the set $\{(c, v) | v \leq V(c)\}$ is a convex set.
- This is an $(m+1)$-dimensional set, the collection of all points (c, v) such that $v \leq V(c)$.
- That is, revenue of v can be produced using the input vector c.

Non-decreasing and Concave $V(c)$

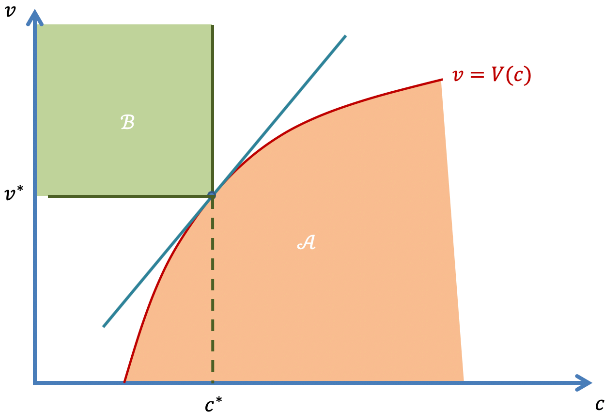


Separation

- Since $\mathcal{A}$ is a convex set, it can be separated from other convex sets.
- Choose a point $(c^*, v^*) \in \mathcal{A}$ such that $v^* = V(c^*)$.
- (c^*, v^*) must be a boundary point since for any $r > 0$,
  (i) $v^* - r < v^* = V(c^*)$ implies that the point $(c^*, v^* - r)$ is in $\mathcal{A}$;
  (ii) $v^* + r > v^* = V(c^*)$ implies that the point $(c^*, v^* + r)$ is not in $\mathcal{A}$.

Separation

- Define $\mathcal{B}$ as the set of all points (c, v) such that $c \leq c^*$ and $v \geq v^*$.



Separation

(i) **Convexity of $\mathcal{B}$.**

- For any two points $(c, v), (c', v') \in \mathcal{B}$:

  $c \leq c^*, v \geq v^*$ and $c' \leq c^*, v' \geq v^*$

  and any real number $\alpha \in [0, 1]$

- we have $\alpha c + (1-\alpha)c' \leq \alpha c^* + (1-\alpha)c^* = c^*$

  $$\alpha v + (1-\alpha)v' \geq \alpha v^* + (1-\alpha)v^* = v^*$$

- That is, $(\alpha c + (1-\alpha)c', \alpha v + (1-\alpha)v') \in \mathcal{B}$.

(ii) **No Common Interior.**

- Points in $\mathcal{A}$ satisfy $v \leq V(c)$.

- For points $(c, v) \in \mathcal{B}$,

$$v \geq v^* = V(c^*) \underbrace{\geq}_{V(c) \text{ is non-decreasing.}} V(c) \implies v \geq V(c).$$

- Therefore, $\mathcal{A}$ and $\mathcal{B}$ do not have interior points in common.

Separation

- We could apply the Separation Theorem.
- (c^*, v^*) is a common boundary point of $\mathcal{A}$ and $\mathcal{B}$.
- We could write the equation of the separating hyperplane as follows: $\iota v - \lambda c = b = \iota v^* - \lambda c^*$, where ι is a scalar, and λ is a m-dimensional row vector.
- The signs are so chosen that

$$\iota v - \lambda c \begin{cases} \leq b & \text{for all } (c, v) \in \mathcal{A} \\ \geq b & \text{for all } (c, v) \in \mathcal{B}. \end{cases} \tag{7.6}$$

Remark. *ι and λ must both be non-negative.*

(i) $\boldsymbol{\iota} \geq \mathbf{0}$:

- Suppose $\iota < 0$.
- The point $(c^*, v^* + 1) \in \mathcal{B}$.
- However, $\iota(v^* + 1) - \lambda c^* = b + \iota < b$, contradicting with

$$\iota v - \lambda c \begin{cases} \leq b & \text{for all } (c, v) \in \mathcal{A} \\ \geq b & \text{for all } (c, v) \in \mathcal{B}. \end{cases} \tag{7.6}$$

Remark. *ι and λ must both be non-negative.*

(ii) $\boldsymbol{\lambda_i \geq 0}$ **for** $\boldsymbol{i = 1, 2, ..., m}$:

- Suppose $\lambda_i < 0$.
- The point $(c^* - e^i, v^*) \in \mathcal{B}$.
- However, $\iota v^* - \lambda(c^* - e^i) = b + \lambda_i < b$, contradicting with

$$\iota v - \lambda c \begin{cases} \leq b & \text{for all } (c, v) \in \mathcal{A} \\ \geq b & \text{for all } (c, v) \in \mathcal{B}. \end{cases} \tag{7.6}$$

Separation

Now comes the more subtle question:

Question. *Can ι be zero?*

Consequence of $\iota = 0$

(i) • For the equation of the hyperplane $\iota v - \lambda c = b$ to be meaningful, the combined vector (ι, λ) must be non-zero.

• Therefore, $\lambda_i \neq 0$ for at least one i.

• Given that $\lambda_i \geq 0$ for all i, $\lambda_i > 0$ for at least one i.

(ii) • The equation of the hyperplane becomes

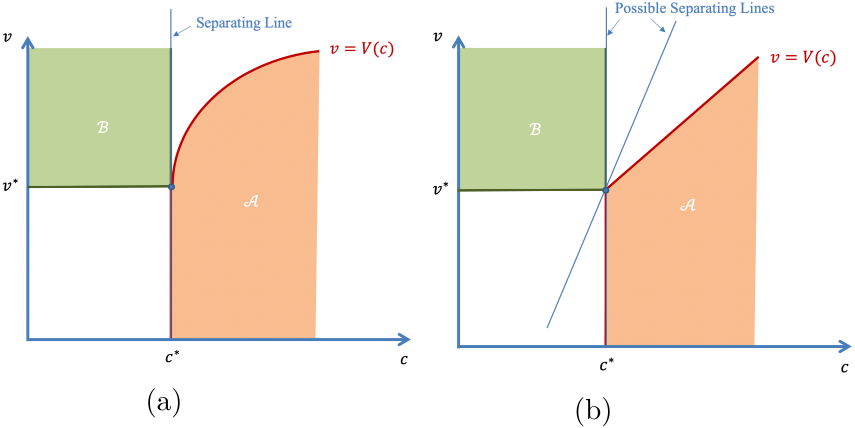
$-\lambda c = b = -\lambda c^*$.

• For all $(c, v) \in \mathcal{A}$, $-\lambda c \leq -\lambda c^*$, or $\lambda(c - c^*) \geq 0$.

Consequence of $\iota = 0$

- In the scalar constraint case, in such a situation, we have $\lambda > 0$.
- Therefore, $\lambda(c - c^*) \geq 0$ implies $c - c^* \geq 0$.
- Graphically, the separating line is vertical at c^*, and the set $\mathcal{A}$ lies entirely to the right of it.

Consequence of $\iota = 0$



Consequence of $\iota = 0$

(i) In case 7.1a, only a vertical separating line exists.

(ii) In case 7.1b, the limit from the right stays finite, and there exists both a vertical separating line and many other non-vertical separating lines. Those non-vertical separating lines are with positive ι.

Therefore, the conditions soon to be found for ensuring a positive ι, which is to ensure the existence of c such that $c < c^*$, are only sufficient but not necessary.

Constraint Qualification

Claim. *If there exists an x^o such that $G(x^o) \ll c^*$ and $F(x^o)$ is defined, then $\iota > 0$.*

- This requirement is the *constraint qualification* for the concave programming problem.

- It is sometimes called the *Slater condition*.

Constraint Qualification: Intuition

- For a scalar c, such a condition works since

  (i) $(G(x^o), F(x^o)) \in \mathcal{A}$ and

  (ii) $(G(x^o), F(x^o))$ is a point to the left of c^*.

  $(G(x^o) < c^*\ )$

- The separating line cannot have an infinite slope at c^*.

Constraint Qualification

Proof. We prove by contradiction.

- Suppose that the condition holds ($G(x^o) \ll c^*$ and $F(x^o)$ defined) but $\iota = 0$.

- On one hand, $\lambda_i \geq 0$ for all i; $\lambda_i > 0$ for at least one i.

- Therefore, by $G(x^o) \ll c^* \iff G^i(x^o) < c_i^*$, we have

$$\Longrightarrow \lambda(G(x^o) - c^*) = \sum_{i=1}^{m} \lambda_i (G^i(x^o) - c_i^*) < 0. \qquad (7.7)$$

Constraint Qualification

- On the other hand, $(G(x^o), F(x^o)) \in \mathcal{A}$ since revenue of $F(x^o)$ can be generated using the input vector $G(x^o)$.
- Therefore, by the separation property,

$$-\lambda G(x^o) \underbrace{=}_{\iota=0} \iota F(x^o) - \lambda G(x^o) \underbrace{\leq}_{\text{separation property}} \iota v^* - \lambda c^* \underbrace{=}_{\iota=0} -\lambda c^*$$

$$\Longrightarrow \lambda(G(x^o) - c^*) \geq 0. \tag{7.8}$$

- (7.8) contradicts

$$\lambda(G(x^o) - c^*) < 0. \tag{7.7}$$

Normalization

- The separation property (7.6) is unaffected if we multiply by b, ι and λ_i by the same positive number.
- Once we can be sure that $\iota \neq 0$, we can choose a scale to make $\iota = 1$.
- In economic terms, ι and λ constitute a system of shadow prices, ι for revenue and λ for the inputs.
- Only relative prices matter for ecoonomic decisions, in setting $\iota = 1$, we are choosing revenue to be the numéraire.
- We will adopt this normalization henceforth.

Shadow Price Interpretation of λ

- Observe that by the separation property (7.6), for all $(c, v) \in \mathcal{A}$,
$$v - \lambda c \leq v^* - \lambda c^*.$$
- That is, (c^*, v^*) achieves the maximum value of $(v - \lambda c)$ among all points $(c, v) \in \mathcal{A}$.
- If we interpret λ as the vector of shadow prices of inputs, then $(v - \lambda c)$ is the profit that accrues when a producer uses inputs c to produce revenue v.

Shadow Price Interpretation of λ

- Since all points in $\mathcal{A}$ represents feasible production plans, the result says that a profit-maximizing producer will pick (c^*, v^*).
- This means that the producer need not be aware that in fact the availability of inputs is limited to c^*.
- He may think that he is free to choose any c but ends up choosing the right c^*.
- It is the prices λ that brings home to him the scarcity.

Shadow Price Interpretation of λ

- The principle behind this interpretation is general and important: *constrained choice can be converted into unconstraint choice if the proper scarcity costs or shadow values of the cosntraints are netted out of the criterion function.*

- As it will become clear later, this is the most important feature of Lagrange's Method in concave programming.

Generalized Marginal Products

- For any c, the point $(c, V(c))$ is in $\mathcal{A}$.
- So by the separation property, we have

$$V(c) - \lambda c \leq V(c^*) - \lambda c^*,$$

$$\text{or} \quad V(c) - V(c^*) \leq \lambda(c - c^*). \tag{7.9}$$

- If $V(c)$ is differentiable, then by Proposition 7.A.1, concavity of $V(c)$ means

$$V(c) - V(c^*) \leq V_c(c^*)(c - c^*). \tag{7.10}$$

- (7.9) and (7.10) suggest $\lambda = V_c(c^*)$ (shadow prices)

Generalized Marginal Products

- However, the problem is that V may not be differentiable.
- Let us consider a general point $(c, V(c))$ with its associated multiplier vector λ.
- Compare this with a neighboring point where only the i^{th} input is increase: $(c + he^i, V(c + he^i))$, where h is a positive scalar and e^i is a vector with its i^{th} component equal to 1 and all others 0.

Generalized Marginal Products

- Then by the separation property

$$V(c) - V(c^*) \leq \lambda(c - c^*). \tag{7.9}$$

we have

$$V(c + he^i) - V(c) \leq \lambda he^i = h\lambda_i$$

$$\Longrightarrow \frac{[V(c + he^i) - V(c)]}{h} \leq \lambda_i. \tag{7.11}$$

- We will show that by the concavity of V, the left-hand side of (7.11) is a non-increasing function of h.

Generalized Marginal Products

- To see this, consider two points $(c+he^i, V(c+he^i))$ and $(c+\alpha he^i, V(c+\alpha he^i))$ for some $h>0$ and $\alpha \in (0,1)$.
- Then by concavity of V,

$$V(c+\alpha he^i)) \geq \alpha V(c+he^i)) + (1-\alpha)V(c)$$

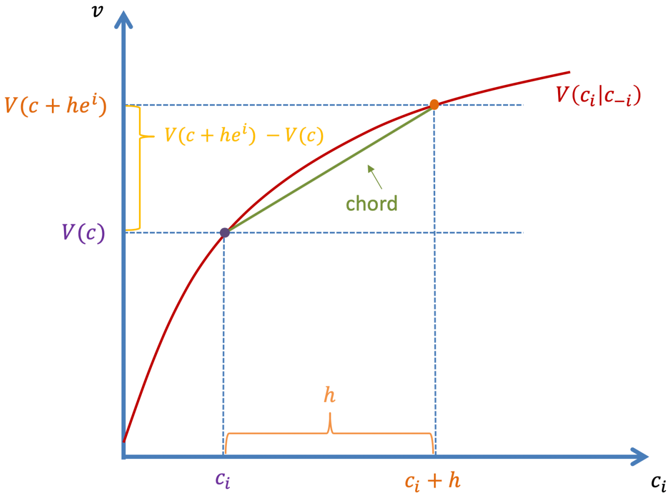
$$\Longrightarrow V(c+\alpha he^i)) - V(c) \geq \alpha \left[V(c+he^i)) - V(c)\right]$$

$$\Longrightarrow \frac{V(c+\alpha he^i)) - V(c)}{\alpha h} \geq \frac{V(c+he^i)) - V(c)}{h} \quad (7.12)$$

- Since $\alpha h < h$, (7.12) implies that the left-hand side of (7.11), namely, $\frac{V(c+he^i))-V(c)}{h}$ is non-increasing in h.

Generalized Marginal Products

Graphically, $\frac{[V(c+he^i)-V(c)]}{h}$ is simply the slope of the chord.



Generalized Marginal Products

- Therefore, the left-hand side expression must attain the maximum as as h goes to zero from positive values.
- This limit is defined as the "rightward" partial derivative of V with respect to the i^{th} coordinate of c: $V_i^+(c)$.
- Therefore,

$$\frac{[V(c+he^i) - V(c)]}{h} \leq \lambda_i. \tag{7.11}$$

implies $V_i^+(c) \leq \lambda_i$.

Generalized Marginal Products

- Similarly, we could repeat the analysis for $h < 0$.
- Now, (7.9) implies

$$\begin{aligned} & V(c + he^i)) - V(c) \leq \lambda he^i = h\lambda_i \\ \Longrightarrow & \frac{[V(c + he^i)) - V(c)]}{h} \geq \lambda_i. \end{aligned} \tag{7.13}$$

- Taking the limit from the negative values of h gives the "leftward" partial derivative $V_i^-(c)$.
- This proves $V_i^-(c) \geq \lambda_i$.

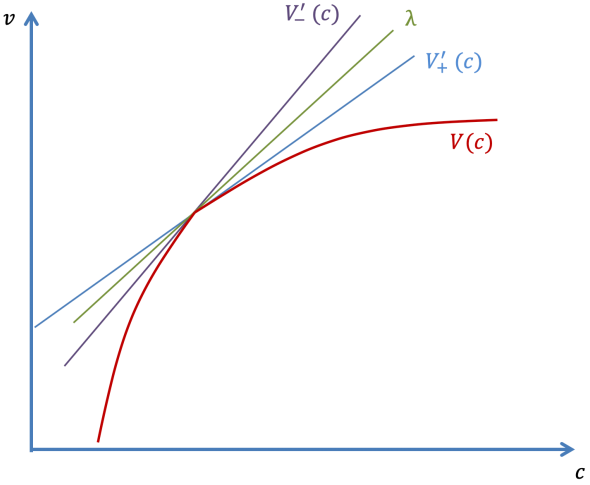
Generalized Marginal Products

- Combining the two, we have

$$V_i^-(c) \geq \lambda_i \geq V_i^+(c). \tag{7.14}$$

- This result generalizes the notion of diminishing marginal returns and relates the multipliers to these generalized marginal products.

Generalized Marginal Products



Choice Variables

- So far the vector of choice variables x has been kept in the background.

- Let's now consider it explicitly.

Choice Variables

- The point $(G(x^*), F(x^*)) \in \mathcal{A}$
- The separation property gives

$$F(x^*) - \lambda G(x^*) \leq V(c) - \lambda c \underbrace{\implies}_{F(x^*)=V(c)} \lambda \left[c - G(x^*)\right] \leq 0.$$

- That is, $\sum_{i=1}^{m} \lambda_i \left[c_i - G^i(x^*)\right] \leq 0.$
- Since $\lambda_i \geq 0$ and $G^i(x) \leq c_i$ for all i, we have

  $\lambda_i \left[c_i - G^i(x^*)\right] \geq 0$ for all i.
- Therefore, $$\lambda_i \left[c_i - G^i(x^*)\right] = 0. \tag{7.15}$$
- This is just *complementary slackness.*

Choice Variables

- For any x, the point $(G(x), F(x)) \in \mathcal{A}$.
- Recognizing $$\lambda_i \left[c_i - G^i(x^*)\right] = 0, \qquad (7.15)$$

  the separation property gives

  $$F(x) - \lambda G(x) \underbrace{\leq}_{\text{separation property}} V(c) - \lambda c \underbrace{=}_{F(x^*)=V(c) \text{ and } (7.15)} F(x^*) - \lambda G(x^*) \text{ for all } x.$$
- x^* maximizes $F(x) - \lambda G(x)$ without any constraints.
- This means that the shadow prices allow us to convert the original constrained revenue-maximization problem into an unconstrained profit-maximization problem.

Necessary Conditions for Concave Programming

Theorem 7.1 (Necessary Conditions for Concave Programming). *Suppose that F is a concave function and G is a vector convex function, and that there exists an x^o satisfying $G(x^o) \ll c$. If x^* maximizes $F(x)$ subject to $G(x) \leq c$, then there is a row vector λ such that*

(i) x^* maximizes $F(x) - \lambda G(x)$ without any constraints, and

(ii) $\lambda \geq 0$, $G(x^*) \leq c$ with complementary slackness.

Necessary Conditions for Concave Programming

- Theorem 7.1 does not require F and G to have derivatives.
- But if the functions are differentiable, then we have the first-order necessary conditions for the maximization problem (i):

$$F_x(x^*) - \lambda G_x(x^*) = 0. \tag{7.16}$$

- In terms of the Lagrangian $\mathcal{L}(x, \lambda)$, (7.16) becomes $\mathcal{L}_x(x^*, \lambda)$.
- This is just the condition of Lagrange's Theorem.
- We could further add the non-negativity constraints on x, and get Kuhn-Tucker Theorem.

Necessary Conditions for Concave Programming

- There is one respect in which concave programming goes beyond the general Lagrange or Kuhn-Tucker conditions.
- The first-order necessary conditions (7.16) are not sufficient to ensure maximum.
- In general, there was no claim that x^* maximized the Lagrangian.
- However, when F is concave and G is convex, part (i) of Theorem 7.1 is easily transformed into $\mathcal{L}(x, \lambda) \leq \mathcal{L}(x^*, \lambda)$ for all x, so x^* does maximize the Lagrangian.

Necessary Conditions for Concave Programming

Our interpretation of Lagrange's method as converting the constrained revenue-maximization into unconstrained profit-maximization must be confined to the case of concave programming.

Sufficient Conditions for Concave Programming

- The first-order necessary conditions are *sufficient* to yield a true maximum in the concave programming problem.
- The argument proceeds in two parts.

Sufficient Conditions for Concave Programming

(i) • Suppose x^* satisfies (i) and (ii) in Theorem 7.1.

• Then, for any feasible x, we have

$$F(x^*) - \lambda G(x^*) \underbrace{\geq}_{\text{(i)}} F(x) - \lambda G(x)$$

$$\Longrightarrow F(x^*) - \lambda c \underbrace{\geq}_{\text{(ii) complementary slackness: } \lambda[c-G(x^*)]=0} F(x) - \lambda G(x)$$

$$\Longrightarrow F(x^*) \geq F(x)) - \lambda[c - G(x)] \underbrace{\geq}_{x \text{ is feasible: } G(x) \leq c} F(x).$$

• Thus, x^* maximizes $F(x)$ subject to $G(x) \leq c$.

Sufficient Conditions for Concave Programming

(ii)
- Suppose x^* satisfies the first-order condition

$$F_x(x^*) - \lambda G_x(x^*) = 0. \tag{7.16}$$

- Since F is concave, G is convex, and $\lambda \geq 0$, then $F - \lambda G$ is concave.

- $$[F(x) - \lambda G(x)] - [F(x^*) - \lambda G(x^*)]$$

$$\underbrace{\leq}_{\text{Proposition 7.A.1: Concavity}} [F_x(x) - \lambda G_x(x)]\,(x - x^*) \underbrace{=}_{(7.16)} 0.$$

- Therefore, $F(x) - \lambda G(x) \leq F(x^*) - \lambda G(x^*)$, or x^* maximizes $F(x) - \lambda G(x)$ without any constraints.

Sufficient Conditions for Concave Programming

Theorem 7.2 (Sufficient Conditions for Concave Programming). *If x^* and λ are such that*

(i) x^* maximizes $F(x) - \lambda G(x)$ without any constraints, and

(ii) $\lambda \geq 0$, $G(x^*) \leq c$ with complementary slackness,

then x^* maximizes $F(x)$ subject to $G(x) \leq c$. If $F - \lambda G$ is concave (for which in turn it suffices to have F concave and G convex), then

$$F_x(x^*) - \lambda G_x(x^*) = 0 \tag{7.16}$$

implies (i) above.

Sufficient Conditions for Concave Programming

Note that no constraint qualification appears in the sufficient conditions.

7.C. Quasi-concave Programming

- In the separation approach of Chapter 6, F was merely quasi-concave and each component constraint function in G was quasi-convex.
- In this chapter, the stronger assumption of concavity and convexity has been made so far.

Quasi-concave Programming

- In fact, the weaker assumptions of quasi-concavity (quasi-convexity) make little difference to necessary conditions.
- They yield sufficient conditions like the ones above for concave programming, but only in the presence of some further technical conditions that are complex to establish.
- For interested students, please refer to the paper "Arrow and Enthoven (1961). Quasi-concave Programming. *Econometrica*, 779-800."

Quasi-concave Programming

We will discuss only a limited version of quasi-concave programming, namely, the one where the objective function is quasi-concave and the constraint function is linear:[1]

$$\max_x F(x) \tag{MP1}$$
$$\text{s.t. } px \le b,$$

where p is a row vector and b is a number.

[1]The mirror-image case of a linear objective and a quasi-convex constraint can be treated in the same way.

Quasi-concave Programming

Recall the definition of Quasiconcavity:

Definition 6.B.3 (Quasi-concave Function). *A function* $f : \mathcal{S} \to \mathbb{R}$*, defined on a convex set* $\mathcal{S} \subset \mathbb{R}^N$*, quasi-concave if the set* $\{x|f(x) \geq c\}$ *is convex for all* $c \in \mathbb{R}$*, or equivalently, if* $f(\alpha x^a + (1-\alpha)x^b) \geq \min\{f(x^a), f(x^b)\}$*, for all* x^a*,* x^b *and for all* $\alpha \in [0, 1]$*.*

Quasi-concave Programming

- For a quasi-concave objective function F, suppose $F(x^b) \geq F(x^a)$.
- Then,
$$F((1-\alpha)x^a + \alpha x^b) \geq F(x^a), \tag{7.17}$$
for all $\alpha \in [0,1]$.
- Let $h(\alpha) = F((1-\alpha)x^a + \alpha x^b) = F(x^a + \alpha(x^b - x^a))$.
- Then, (7.17) becomes
$$h(\alpha) \geq h(0) \implies \frac{h(\alpha) - h(0)}{\alpha} \geq 0. \tag{7.18}$$

- By the definition of derivative,

$$\lim_{\alpha \to 0} \left[\frac{h(\alpha) - h(0)}{\alpha} \right] = h'(0).$$

- Since

$$\frac{h(\alpha) - h(0)}{\alpha} \geq 0 \tag{7.18}$$

holds when $\alpha \to 0$, we have

$$h'(0) \geq 0. \tag{7.19}$$

Quasi-concave Programming

- On the other hand, by chain rule,

$$h'(\alpha) = F_x(x^a + \alpha(x^b - x^a))(x^b - x^a)$$

$$\Longrightarrow h'(0) = F_x(x^a)(x^b - x^a) \tag{7.20}$$

- Together with

$$h'(0) \geq 0. \tag{7.19}$$

we have

$$F_x(x^a)(x^b - x^a) \geq 0. \tag{7.21}$$

- This holds for all x^a, x^b such that $F(x^b) \geq F(x^a)$.

Quasi-concave Programming

- Now consider the maximization problem

$$\max_x F(x) \tag{MP1}$$

$$\text{s.t. } px \leq b,$$

- The first-order necessary conditions are

$$F_x(x^*) - \lambda p = 0 \tag{7.22}$$

$$px^* \leq b \text{ and } \lambda \geq 0, \text{ with complementary slackness}$$

Quasi-concave Programming

We claim that (7.22) is also sufficient when $\lambda > 0$ and the constraint is binding.[2] Formally,

Claim. *If F is continuous and quasi-concave, x^* and $\lambda > 0$ satisfy the first-order necessary conditions, then x^* solves the quasi-concave programming problem.*

[2]Appendix B provides an example of a spurious stationary point where (7.22) holds with $\lambda = 0$.

Proof. We prove by contradiction.

- Suppose that there exists x such that $F(x) > F(x^*) \equiv v^*$.

- We will show that x is not feasible, that is, $px > b$.

Quasi-concave Programming

- By
$$F_x(x^a)(x^b - x^a) \geq 0, \tag{7.21}$$

$$F(x) > F(x^*) \text{ implies } \quad F_x(x^*)(x - x^*) \geq 0. \tag{7.23}$$

- Substituting
$$F_x(x^*) - \lambda p = 0 \tag{7.22}$$

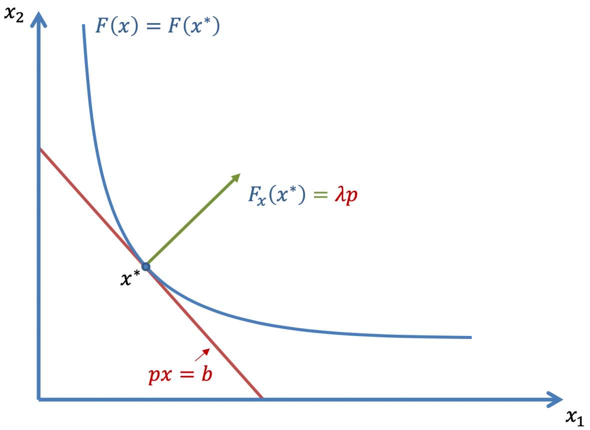
$$\lambda p(x - x^*) \geq 0 \underbrace{\Longrightarrow}_{\lambda > 0} p(x - x^*) \geq 0 \quad \text{or} \quad px \geq px^* \underbrace{=}_{\text{constraint binding}} b.$$

- In other words, the upper contour set of $F(x)$ for the value v^* is contained in the half-space on or above the constraint line.

Quasi-concave Programming

- Since F is continuous and $F(x) > F(x^*)$, x is an interior point of the upper contour set of $F(x)$ for the value v^*.
- Therefore, it is also an interior point of the set $px \geq b$.
- In other words, it satisfies $px > b$. $\square$

Quasi-concave Programming



Quasi-concave Programming

- $F_x(x^*)$ is normal to the contour of $F(x)$ at x^*.
- p is normal to the constraint $px = b$ at x^*.
- The usual tangency condition is equivalent to the normal vectors being parallel.
- Equation (7.22) expresses this, with the constant of proportionality equal to λ.

7.D. Uniqueness

- The above sufficient conditions for concave as well as quasi-concave programming are *weak* in the sense that they establish that no other feasible choice x can do better than x^*.

- They do not rule out the existence of other feasible choices that yield $F(x) = F(x^*)$.

- In other words, they do not establish the uniqueness of the optimum.

Uniqueness

As discussed in Chapter 6, a strenghening of the concept of concavity or quasi-concavity gives uniqueness.

Definition 7.D.1 (Strictly Concave Function)**.** *A function* $f : \mathcal{S} \to \mathbb{R}$*, defined on a convex set* $\mathcal{S} \subset \mathbb{R}^N$*, is strictly concave if*

$$f(\alpha x^a + (1-\alpha)x^b) > \alpha f(x^a) + (1-\alpha)f(x^b), \qquad (7.24)$$

for all x^a, $x^b \in \mathcal{S}$ *and for all* $\alpha \in (0,1)$.

Uniqueness

Claim. *If the objective function F in the concave programming problem is strictly concave, then the maximizer x^* is unique.*

Uniqueness

Proof. We prove by contradiction.

- Suppose that $x^{*\prime}$ is another solution.
- Then, $F(x^*) = F(x^{*\prime}) = v^*$, and $G(x^*) \leq c$, $G(x^{*\prime}) \leq c$.
- Now consider $\alpha x^* + (1-\alpha)x^{*\prime}$.

Uniqueness

(i) $\alpha x^* + (1-\alpha)x^{*\prime}$ is feasible since for each i,

$$G^i(\alpha x^* + (1-\alpha)x^{*\prime}) \underbrace{\leq}_{\text{convexity}} \alpha G^i(x^*) + (1-\alpha)G^i(x^{*\prime})$$

$$\underbrace{\leq}_{\text{feasibility of } x^* \text{ and } x^{*\prime}} \alpha c_i + (1-\alpha)c_i = c_i.$$

(ii) $\alpha x^* + (1-\alpha)x^{*\prime}$ yields higher value than v^* since

$$F(\alpha x^* + (1-\alpha)x^{*\prime}) \underbrace{>}_{\text{strict concavity}} \alpha F(x^*) + (1-\alpha)F(x^{*\prime})$$

$$= \alpha v^* + (1-\alpha)v^* = v^*.$$

Uniqueness

- Therefore, we have found a feasible choice $\alpha x^* + (1 - \alpha x^{*\prime})$ which yields higher value than v^*.

- This contradicts with the fact the x^* and $x^{*\prime}$ are optimal.

- Therefore, the initial supposition must be wrong and strict concavity of F implies the uniqueness of the maximizer.

7.E. Examples

Example 7.1: Linear Programming

An important special case of concave programming is the theory of *linear programming.*

$$\max_x F(x) \equiv ax \tag{Primal}$$

$$\text{s.t. } G(x) \equiv Bx \leq c \text{ and } x \geq 0,$$

where a is an n-dimensional row vector and B an m-by-n matrix.

Example 7.1: Linear Programming

- Now

$$F_x(x) = a \text{ and } G_x(x) = B.$$

- When the constraint functions are linear, no constraint qualification is needed.

- All conditions of concave programming are fulfilled, and the Kuhn-Tucker conditions are both necessary and sufficient.

Example 7.1: Linear Programming

- The Lagrangian is

$$\mathcal{L}(x, \lambda) = ax + \lambda[c - Bx]. \tag{7.25}$$

- The optimum x^* and λ^* satisfy Kuhn-Tucker conditions:

$$a - \lambda^* B \leq 0,\ x^* \geq 0, \text{with complementary slackness,} \tag{7.26}$$

$$c - Bx^* \geq 0,\ \lambda^* \geq 0, \text{with complementary slackness.} \tag{7.27}$$

Example 7.1: Linear Programming

- (7.26) and (7.27) contain 2^{m+n} combinations of patterns of equations and inequalities.

- As a special feature of the linear programming problem, if k of the constraints in (7.27) hold with equality, then exactly $(n-k)$ non-negativity constraints in (7.26) should bind.

- When this is the case, the corresponding equations for λ is also of the correct number m.

Example 7.1: Linear Programming

Next, consider a new linear programming problem:

$$\max_y -yc \tag{Dual}$$

$$\text{s.t.} \quad -yB \leq -a \text{ and } y \geq 0,$$

where y is a m-dimensional row vector and the vectors a, c and the matrix B are exactly as before.

Example 7.1: Linear Programming

- We introduce a column vector μ of multipliers and define the Lagrangian:

$$\mathcal{L}(x, \lambda) = -yc + [-a + yB]\mu. \tag{7.28}$$

- The optimum y^* and μ^* satisfy the necessary and sufficient Kuhn-Tucker conditions:

$$-c + B\mu^* \leq 0,\ y^* \geq 0, \text{with complementary slackness,} \tag{7.29}$$

$$-a + y^*B \geq 0,\ \mu^* \geq 0, \text{with complementary slackness.} \tag{7.30}$$

Example 7.1: Linear Programming

- (7.29) is exactly the same as (7.27) and (7.30) is exactly the same as (7.26), if we replace y^* by λ^* and μ^* by x^*.
- In other words, the optimum x^* and λ^* solve the new problem.
- The new problem is said to be *dual* to the original, which is then called the *primal* problem in the pair.
- This captures an important economic relationship between prices and quantities in economics.

Example 7.1: Linear Programming

- We interpret the primal problem as follows:

$$\max_x \underbrace{a}_{\text{output prices}} \underbrace{x}_{\text{output quantities}}$$

$$\text{s.t. } \underbrace{Bx}_{\text{inputs for producing } x} \leq \underbrace{c}_{\text{input supplies}} \text{ and } x \geq 0,$$

- Solving the primal problem, we get x^* and λ^*.

- λ^* is the vector of shadow prices of the inputs.

Example 7.1: Linear Programming

- Rewriting the dual problem in terms of λ.
- We know from the previous analysis that λ^* solves the dual problem.

$$\lambda^* = \min_{\lambda}\{\lambda c \mid \lambda B \geq a \text{ and } \lambda \geq 0\}$$

- Thus, the shadow prices minimize the cost of the input c.

Example 7.1: Linear Programming

- Note that the j^{th} component of λB is $\sum_i \lambda_i B_{ij}$, which is the cost of the bundle of inputs needed to produce one unit of good j, calcualted using the shadow prices.
- The constraint $\sum_i \lambda_i B_{ij} \geq a_j$ means that the input cost of good j is at least as great as the unit value of output of good j. This is true for all good j.
- In other words, the shadow prices of inputs ensure that no good can make a strictly positive profit – a standard "competitive" condition in economics.

Example 7.1: Linear Programming

Complementary slackness in (7.26) ensures that

(i) If the unit cost of production of j, $\sum_i \lambda_i B_{ij}$, exceeds its prices a_j, then $x_j = 0$. That is, if the production of j would entail a loss when calculated using the shadow prices, then good j would not be produced.

(ii) If good j is produced in positive quantity, $x_j > 0$, then the unit cost exactly equals the price, $\sum_i \lambda_i B_{ij} = a_j$. That is, the profit is exactly 0.

Example 7.1: Linear Programming

- Complementary slackness in (7.26) and (7.27) imply

$$[a - \lambda^* B]x^* = 0 \implies ax^* = \lambda^* B x^*$$

$$\lambda^*[c - Bx^*] = 0 \implies \lambda^* c = \lambda^* B x^*$$

- Combining the two, we have $ax^* = \lambda^* c \qquad (7.31)$
- This says that the value of the optimum output equals the cost of the factor supplies.
- This result can be interpreted as the circular flow of income, that is, national product equals national income.

Example 7.1: Linear Programming

- Finally, it is easy to check that if we take the dual problem as our starting-point and go through the mechanical steps to finding its dual, we return to the primal.
- In other words, duality is reflexive.

Example 7.1: Linear Programming

- This is the essence of the duality theory of linear programming.
- One final remark is that we took the optimum x^* as our starting point, however, the solution may not exist, because the constraints may be mutually inconsistent, or they may define an unbounded feasible set.
- This issue beyond our discussion here and is left to more advanced texts.

Example 7.2: Failure of Profit-maximizing

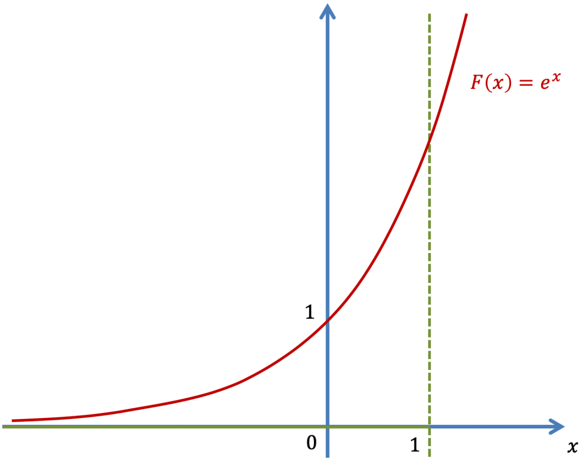
For a scalar x, consider the following maximization problem:

$$\max_x F(x) \equiv e^x$$

$$\text{s. t. } G(x) \equiv x \leq 1.$$

$F(x)$ is increasing, and the maximum occurs at $x = 1$.

Example 7.2: Failure of Profit-maximizing



Example 7.2: Failure of Profit-maximizing

- Kuhn-Tucker Theorem applies.
- The Lagrangian is

$$\mathcal{L}(x, \lambda) = e^x + \lambda(1 - x).$$

- Kuhn-Tucker necessary conditions are

$\partial\mathcal{L}/\partial x = e^x - \lambda = 0;$

$\partial\mathcal{L}/\partial\lambda = 1 - x \geq 0$ and $\lambda \geq 0,$ with complementary slackness.

- The solution is $x^* = 1$ and $\lambda = e$.

Example 7.2: Failure of Profit-maximizing

- However, $x = 1$ does not maximize $F(x) - \lambda G(x)$ without constraints.

- In fact, $e^x - ex$ can be made arbitrarily large by increasing x beyond 1.

- Here, Lagrange's method does not convert the original constrained maximization problem into an unconstrained profit-maximization problem, because F is not concave.