Nonlinear Programming
and the
Kuhn-Tucker Conditions

The Kuhn-Tucker (KT) conditions are first-order conditions for constrained optimization problems, a generalization of the first-order conditions we're already familiar with. These more general conditions provide a unified treatment of constrained optimization, in which

- we allow for inequality constraints;
- there may be any number of constraints;
- constraints may be binding or not binding at the solution;
- non-negativity constraints can be included;
- boundary solutions (some x_i's $= 0$) are permitted;
- non-negativity and structural constraints are treated in the same way;
- dual variables (also called Lagrange multipliers) are shadow values (*i.e.*, marginal values).

A special case covered by the Kuhn-Tucker conditions is linear programming. The conditions are also called the Karush-Kuhn-Tucker conditions: many years after Kuhn and Tucker developed the conditions in 1951, it was discovered that William Karush had presented essentially the same conditions in his 1939 master's degree thesis.

The Kuhn-Tucker Conditions

Let $f : \mathbb{R}^n \to \mathbb{R}$ and $G : \mathbb{R}^n \to \mathbb{R}^m$ be continuously differentiable functions, and let $\mathbf{b} \in \mathbb{R}^m$. We want to characterize those vectors $\widehat{\mathbf{x}} \in \mathbb{R}^n$ that satisfy

$(*)$ $\widehat{\mathbf{x}}$ is a solution of the problem

(P) Maximize $f(\mathbf{x})$ subject to $\mathbf{x} \geqq \mathbf{0}$ and $G(\mathbf{x}) \leqq \mathbf{b}$,

i.e., subject to $x_1, x_2, \ldots, x_n \geqq 0$ and to $G^i(\mathbf{x}) \leqq b_i$ for $i = 1, \ldots, m$.

The **Kuhn-Tucker conditions** are the first-order conditions that characterize the vectors $\widehat{\mathbf{x}}$ that satisfy $(*)$ (when appropriate second-order conditions are satisfied, which we'll see momentarily):

$\exists\, \lambda_1, \ldots, \lambda_m \in \mathbb{R}_+$ such that

(KT1) For $j = 1, \ldots, n$: $\dfrac{\partial f}{\partial x_j} \leqq \sum_{i=1}^m \lambda_i \dfrac{\partial G^i}{\partial x_j}$, with equality if $\widehat{x}_j > 0$;

(KT2) For $i = 1, \ldots, m$: $G^i(\widehat{\mathbf{x}}) \leqq b_i$, with equality if $\lambda_i > 0$,

where the partial derivatives are evaluated at $\widehat{\mathbf{x}}$. The scalars λ_i are called **Lagrange multipliers**.

The Kuhn-Tucker conditions given above are in *partial derivative form*. An equivalent statement of the conditions is in *gradient form*:

$$\begin{array}{l}\exists\, \lambda \in \mathbb{R}^m_+ \text{ such that} \\ \quad \text{(KT1)} \quad \nabla f \leqq \sum_{i=1}^m \lambda_i \nabla G^i \quad \text{and} \quad \widehat{\mathbf{x}} \cdot (\nabla f - \sum_{i=1}^m \lambda_i \nabla G^i) = 0 \ ; \\ \quad \text{(KT2)} \quad G(\widehat{\mathbf{x}}) \leqq \mathbf{b} \quad \text{and} \quad \lambda \cdot (\mathbf{b} - G(\widehat{\mathbf{x}})) = 0 \ , \end{array}$$

where gradients are evaluated at $\widehat{\mathbf{x}}$.

The Kuhn-Tucker Theorems

The first theorem below says that the Kuhn-Tucker conditions are *sufficient* to guarantee that $\widehat{\mathbf{x}}$ satisfies $(*)$, and the second theorem says that the Kuhn-Tucker conditions are *necessary* for $\widehat{\mathbf{x}}$ to satisfy $(*)$. Taken together, the two theorems are called the Kuhn-Tucker Theorem.

Theorem 1: Assume that each G^i is quasiconvex; that either (a) f is concave or (b) f is quasiconcave and $\nabla f \neq \mathbf{0}$ at $\widehat{\mathbf{x}}$; and that f and each G^i are differentiable. If $\widehat{\mathbf{x}}$ satisfies the Kuhn-Tucker conditions then $\widehat{\mathbf{x}}$ satisfies $(*)$.
[Briefly, (KT) $\Rightarrow$ $(*)$.]

Theorem 2: Assume that f is quasiconcave; that each G^i is quasiconvex and the constraint set $\{\mathbf{x} \in \mathbb{R}^n \mid G(\widehat{\mathbf{x}}) \leqq \mathbf{b}\}$ satisfies one of the constraint qualifications (to be described shortly); and that f and each G^i are differentiable. If $\widehat{\mathbf{x}}$ satisfies $(*)$ then $\widehat{\mathbf{x}}$ satisfies the Kuhn-Tucker conditions.
[Briefly, $(*)$ $\Rightarrow$ (KT).]

The next theorem tells us how changes in the values of the b_i's affect the value of the objective function f. For the nonlinear programming problem defined by f, G, and $\mathbf{b}$, define the **value function** $v : \mathbb{R}^m \to \mathbb{R}$ as follows:

$$\forall\, \mathbf{b} \in \mathbb{R}^m : \ v(\mathbf{b}) \text{ is the value of } f(\widehat{\mathbf{x}}) \text{ where } \widehat{\mathbf{x}} \text{ satisfies } (*).$$

Theorem 3: If $(*)$ and (KT) are both satisfied at $\widehat{\mathbf{x}}$, then $\lambda_i = \dfrac{\partial v}{\partial b_i}$ for each i.

In other words, λ_i is the "shadow value" of the i^{th} constraint, the marginal value to the objective function of relaxing or tightening the constraint by one unit.

Note that second-order (curvature/convexity/concavity) conditions are required in order for the Kuhn-Tucker (first-order) conditions to be either necessary or sufficient for $\widehat{\mathbf{x}}$ to be a solution to the nonlinear programming problem.

Examples

Example 1: Suppose that $x_j > 0$ for each $j = 1, \ldots, n$ and that (KT) is satisfied at $\widehat{\mathbf{x}}$. Then

$$\nabla f = \sum_{i=1}^{m} \lambda_i \nabla G^i \text{ for some } \lambda_1, \ldots, \lambda_m \geqq 0,$$

where the gradients are all evaluated at $\widehat{\mathbf{x}}$. In other words, ∇f lies in the cone formed by the gradients of the constraints (it's a non-negative linear combination of them), as in Figure 1.

Example 2: In Example 1 each of the constraints is **binding** — *i.e.*, $G^i(\widehat{\mathbf{x}}) = b_i$, ($\widehat{\mathbf{x}}$ lies *on* each of the constraints) — and each λ_i is strictly positive. The KT conditions allow, however, that if $G^i(\widehat{\mathbf{x}}) = b_i$, then λ_i may be zero, as in Figure 2. This still leaves ∇f in the cone formed by the G^i-gradients ∇G^i.

Example 3: Now suppose that in Figure 2 ∇f and/or ∇G^2 in Figure 2 is perturbed slightly, so that we have Figure 3. Now ∇f no longer lies in the cone of the ∇G^is — ∇f is no longer a *non-negative* linear combination of the G^i-gradients. Notice that there is also a "lens"-shaped region between the f and G^i level curves through $\widehat{\mathbf{x}}$, and that this lens contains points $\mathbf{x}$ that simultaneously satisfy both $f(\mathbf{x}) > f(\widehat{\mathbf{x}})$ and $G^i(\mathbf{x}) \leqq b_i$ for each i — points that are feasible and give larger values of f than $\widehat{\mathbf{x}}$ does. In other words, $\widehat{\mathbf{x}}$ no longer satisfies $(*)$, just as it no longer satisfies (KT).

This idea of the "lens" between level curves — in fact, we do call it a lens — is a useful idea. When f is quasiconcave and each G^i is quasiconvex, the condition that ∇f lies in the cone of the ∇G^is is precisely that any lens between the f-contour and a G^i-contour must lie "outside" of some other G^i-contour — *i.e.*, that the f-contour is tangent (in a generalized sense) to the feasible set at $\widehat{\mathbf{x}}$, and therefore that f is maximized at $\widehat{\mathbf{x}}$.

At the solutions in each of our examples so far, the variables x_j have all been positive and the constraints have all been binding. Examples 4 and 5 have a non-binding constraint, and then a solution at which a variable is zero.

Example 4: If $G^i(\widehat{\mathbf{x}}) < b_i$, then (KT) requires that $\lambda_i = 0$ — *i.e.*, that ∇f lies in the cone formed by the *other* constraints' gradients. This is because, as in Figure 4, a non-binding constraint contributes nothing to defining the feasible set at $\widehat{\mathbf{x}}$, and therefore any lens between the f-contour and any other G^k-contour need only lie outside the set formed by just those other contours.

Example 5: Suppose that $\widehat{x}_2 = 0$, as in Figure 5. Then (KT) allows that $\frac{\partial f}{\partial x_2} < \sum_{i=1}^{m} \lambda_i \frac{\partial G^i}{\partial x_2}$. Now we don't have ∇f in the cone of the G^i-gradients, so there is a lens between the f-contour and one of the G^i-contours that lies inside all the G^i-contours. But the feasible set is truncated by the inequality constraint $x_2 \geqq 0$, so the lens is not actually in the feasible set after all. Analytically, this is captured by the fact that it's $\widehat{x}_2$ that is zero and it's the x_2-component in which equality fails in the inequality $\nabla f \leqq \sum \lambda_i \nabla G^i$. **Important:** Since $\widehat{x}_1 > 0$, (KT) still requires that $\frac{\partial f}{\partial x_1} = \sum_{i=1}^{m} \lambda_i \frac{\partial G^i}{\partial x_1}$ at $\widehat{\mathbf{x}}$. So we have to find λ_i s for which some non-negative linear combination of the binding constraints' gradients lies *directly above* ∇f, as in Figure 5. It must be *above* and not below, because of the strict inequality for the x_2-components of the partial derivatives.

Example 6: Figure 6 shows how the (KT) conditions, which are first-order conditions, can fail to be a sufficient condition for $(*)$ if the second-order curvature conditions don't hold. Here the constraint is not quasiconvex, (KT) is satisfied, but $\widehat{\mathbf{x}}$ is not a solution of the problem **(P)**. You can reinterpret the figure as f not being quasiconcave by pointing the gradients ∇f and ∇G in the opposite direction, to the northwest, and reversing the f and G labels on the level curves.

Example 7: Figure 7 shows how the (KT) conditions can fail to be a necessary condition for $(*)$, even if the second-order curvature conditions *do* hold. Here the two G^i-gradients are co-linear and point in opposite directions, so the two constraints are tangent to one another, and the feasible set consists *only* of the point $\widehat{\mathbf{x}}$, which is therefore trivially a solution of our maximization problem. But ∇f is not in the (degenerate) cone of the G^i-gradients — it is not a linear combination of ∇G^1 and ∇G^2: (KT) is not satisfied.

Constraint Qualifications

In order to ensure that the KT conditions are necessary at a solution, we need to rule out situations like we've just seen in Example 7, where the constraint set was a singleton and the gradients of the constraint functions were linearly dependent. Various *constraint qualifications* have been introduced to accomplish this. The various constraint qualifications are generally independent of one another — *i.e.*, any one of them typically doesn't imply any of the others — and any one of them is enough to ensure that our Theorem 2 is true.

Here are the two constraint qualifications that are by far the most commonly used. They're both obviously violated in Example 7.

Nondegenerate Constraint Qualification (NDCQ): The gradients of the binding constraints (including the binding non-negativity constraints) at the vector $\mathbf{x} \in \mathbb{R}^n$ are linearly independent.

If NDCQ holds at a solution $\widehat{\mathbf{x}}$, and the second-order conditions of Theorem 2 hold, then Theorem 2 says that the KT conditions hold at $\widehat{\mathbf{x}}$. NDCQ therefore requires us to know something about the particular vector $\widehat{\mathbf{x}}$; but of course, when using it for Theorem 2 we know that $\widehat{\mathbf{x}}$ is a solution of the problem **(P)**.

For the other commonly used constraint qualification we need only to establish that there is some vector that satisfies all the constraints as strict inequalities:

Slater's Condition: The constraint set has a nonempty interior — *i.e.*, there is a point $\mathbf{x}$ that satisfies all the constraints as strict inequalities:

$$\forall j = 1, \ldots, n : x_j > 0 \;\text{ and }\; \forall i = 1, \ldots, m : G^i(\mathbf{x}) < b_i.$$

The Lagrangian

Nearly every graduate microeconomics textbook and mathematics-for-economists textbook introduces the **Lagrangian function**, or simply the **Lagrangian**, for constrained optimization problems. For our problem **(P)** the Lagrangian is the function

$$\mathcal{L}(\mathbf{x}, \lambda) := f(\mathbf{x}) + \sum_{i=1}^{m} \lambda_i \big[b_i - G^i(\mathbf{x})\big].$$

The Lagrangian is a way to convert the constrained problem to an unconstrained problem in which the first-order conditions are the Kuhn-Tucker conditions for the original, constrained problem. In order for the Lagrangian first-order conditions to coincide with (KT) we must be at a point $(\widehat{\mathbf{x}}, \widehat{\lambda})$ that maximizes the function $\mathcal{L}(\mathbf{x}, \widehat{\lambda})$ and *minimizes* the function $\mathcal{L}(\widehat{\mathbf{x}}, \lambda)$ and satisfies the inequality constraints $x_j \geqq 0$, $j = 1, \ldots, n$ — in which case (assuming second-order conditions are satisfied) $(\widehat{\mathbf{x}}, \widehat{\lambda})$ will satisfy the Lagrangian's first-order conditions

$$\text{for } j = 1, \ldots, n : \quad \frac{\partial \mathcal{L}}{\partial x_j} \;\leqq\; 0, \qquad \text{with equality if } \widehat{x}_j > 0 \;;$$

$$\text{for } i = 1, \ldots, m : \quad \frac{\partial \mathcal{L}}{\partial \lambda_i} \;\geqq\; 0, \qquad \text{with equality if } \widehat{\lambda}_i > 0 \;;$$

where the partial derivatives are evaluated at $(\widehat{\mathbf{x}}, \widehat{\lambda})$. Since

$$\frac{\partial \mathcal{L}}{\partial x_j} = \frac{\partial f}{\partial x_j} - \sum_{i=1}^{m} \lambda_i \frac{\partial G^i}{\partial x_j} \qquad \text{and} \qquad \frac{\partial \mathcal{L}}{\partial \lambda_i} = b_i - G^i(\mathbf{x}),$$

it's easy to see that these first-order conditions coincide with the Kuhn-Tucker conditions.

While the Lagrangian has other uses, its use for constrained optimization problems seems mostly to be as a way to obtain the Kuhn-Tucker conditions if you can't remember them, or as a way (I think an unsatisfactory way) to *teach* the Kuhn-Tucker conditions.

The other uses of the Lagrangian generally come from the fact that a solution of the Lagrangian maximization/minimization problem is what's called a **saddle-point** of the function $\mathcal{L}$. The problem of identifying such saddle-points — points that maximize with respect to some variables and minimize with respect to the other variables – is called a **saddle-point problem**. To see why such a point is called a saddle-point, consider the graph, in $\mathbb{R}^3$, of the function $f(x, y) = x^2 - y^2$ defined on $\mathbb{R}^2$: at the point $(0, 0)$ the graph looks just like a saddle, as in Figure 8. The saddle-point terminology isn't very descriptive for the Lagrangian, however, because the Lagrangian is linear in the λ variables. Nonetheless, a solution for the Lagrangian function *is* a saddle-point, because it does maximize with respect to the x_j's and minimize with respect to the λ_i's.

The Lagrangian is also a way to remember the Kuhn-Tucker Theorem 3, above. Since $\frac{\partial \mathcal{L}}{\partial b_i} = \lambda_i$, this suggests that the partial derivatives of the value function are the Lagrange multipliers: $\frac{\partial v}{\partial b_i} = \lambda_i$. We'll treat this more formally when we study the Envelope Theorem.

Because the Lagrange multipliers λ_i are the values of the derivatives $\frac{\partial v}{\partial b_i}$, they're the marginal values, or **shadow values**, of the constraints. The multiplier λ_i tells us the value, in objective-function units, of "relaxing" the i-th constraint by one unit. Here it's helpful to think of the x_j variables as the levels at which each of n activities is operated; to think of $f(\mathbf{x})$ as the dollar value — say, the profit — of operating at the levels described by the vector $\mathbf{x}$; and to think of each function $G^i(\mathbf{x})$ as the amount of some resource R_i required to operate at the vector $\mathbf{x}$; and the number b_i as the amount of resource R_i that's available to the decision-maker. Then $\frac{\partial v}{\partial b_i}$ is the marginal increase in profit $f(\mathbf{x})$ that will result from having one additional unit of resource R_i — the marginal value of the resource. If constraint i is non-binding, then the marginal value of an increase or decrease in b_i is zero — *i.e.*, $\lambda_i = 0$. If constraint i is binding, λ_i tells us how much $f(\mathbf{x})$ will be increased or decreased by a unit increase or decrease in availability of resource R_i.

This provides a useful interpretation of the Lagrangian: each λ_i is a dollar reward the decision-maker will receive for every unit of "slack" in constraint i — the amount $b_i - G^i(\mathbf{x})$ — so that the sum $\sum_{i=1}^m \lambda_i \big[b_i - G^i(\mathbf{x})\big]$ is the total reward she will receive, and $\mathcal{L}(\mathbf{x}, \lambda)$ is the total of profit plus reward. But the decision-maker can choose only the activity levels x_j; the scalars λ_i — the dollars per unit of slack the decision-maker will receive as a reward — are chosen by the person who will provide the reward. And that person wants to choose the values of the λ_i's that will make the total reward as *small* as possible. The solution of the Lagrangian problem — the saddle-value — is the combination of $\mathbf{x}$-choices and λ-choices in which each person is optimizing, given the other person's choices. The per-unit rewards λ_i are the marginal values of the resources. And the **complementary slackness** condition $\lambda \cdot \big(\mathbf{b} - G(\mathbf{x})\big) = 0$ in (KT2) says that the total reward will actually be zero at the solution.

FIGURE 1

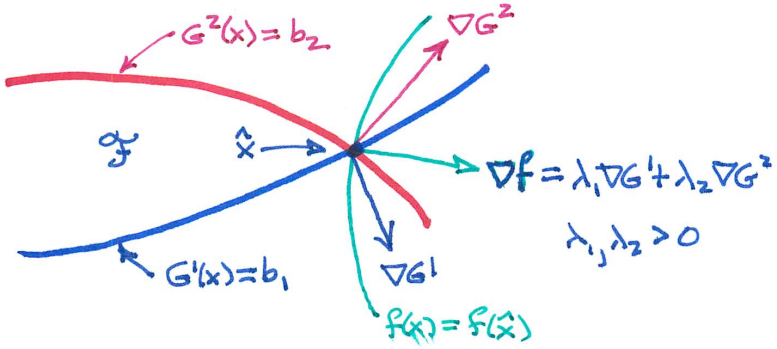


FIGURE 2

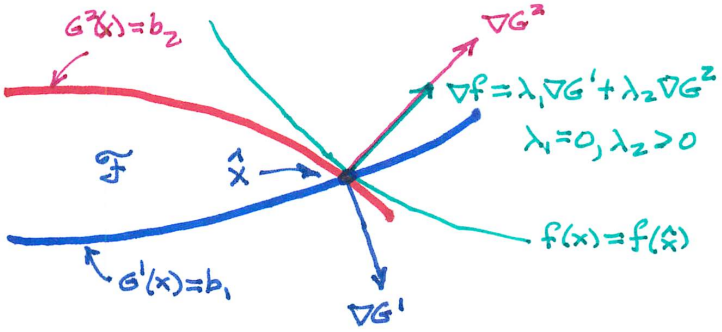


FIGURE 3

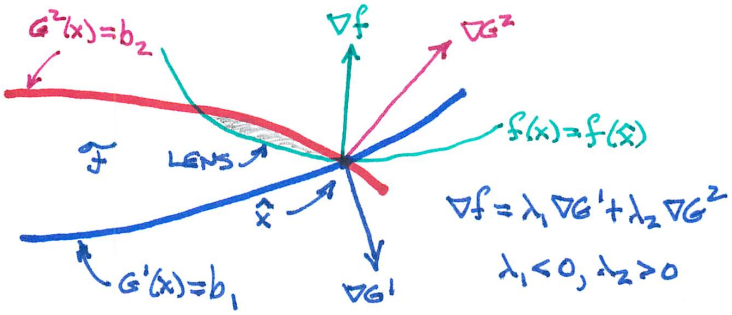


FIGURE 4

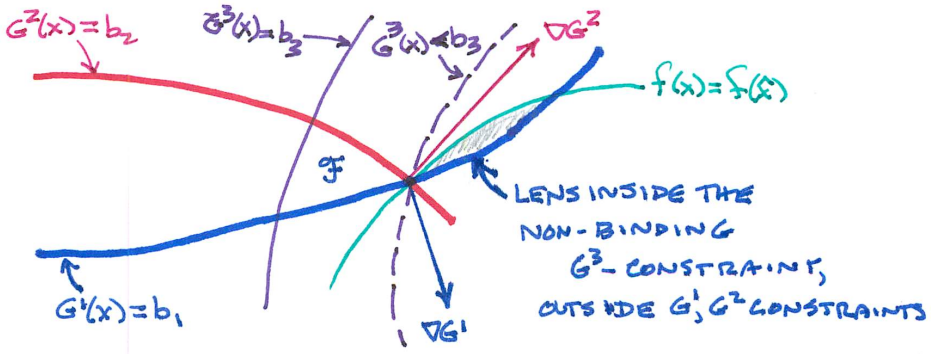


FIGURE 5

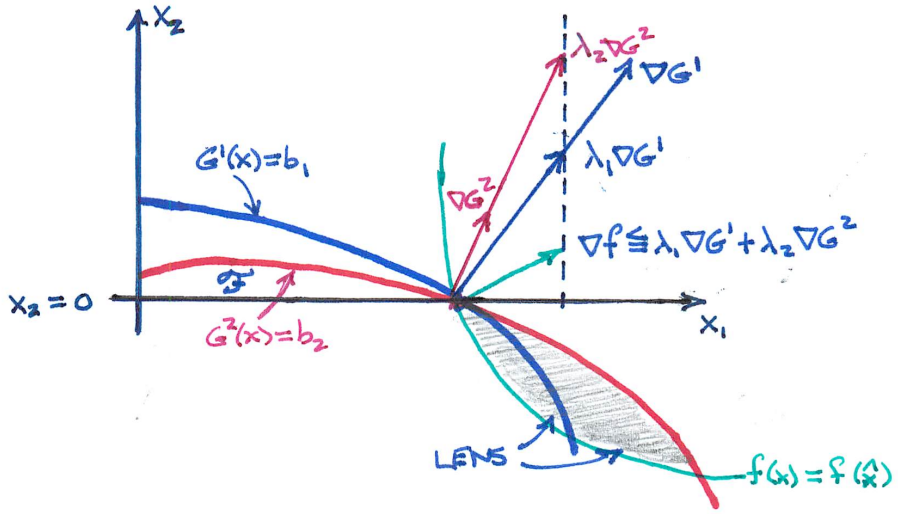


FIGURE 6

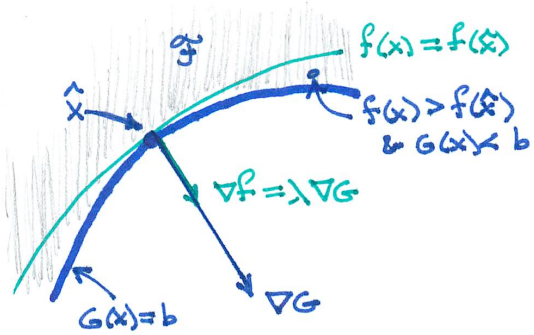
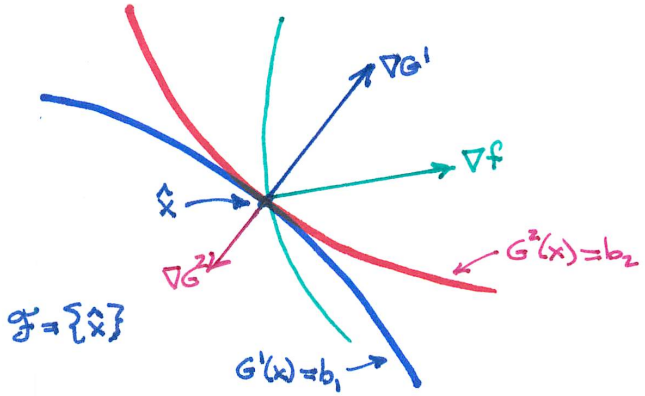
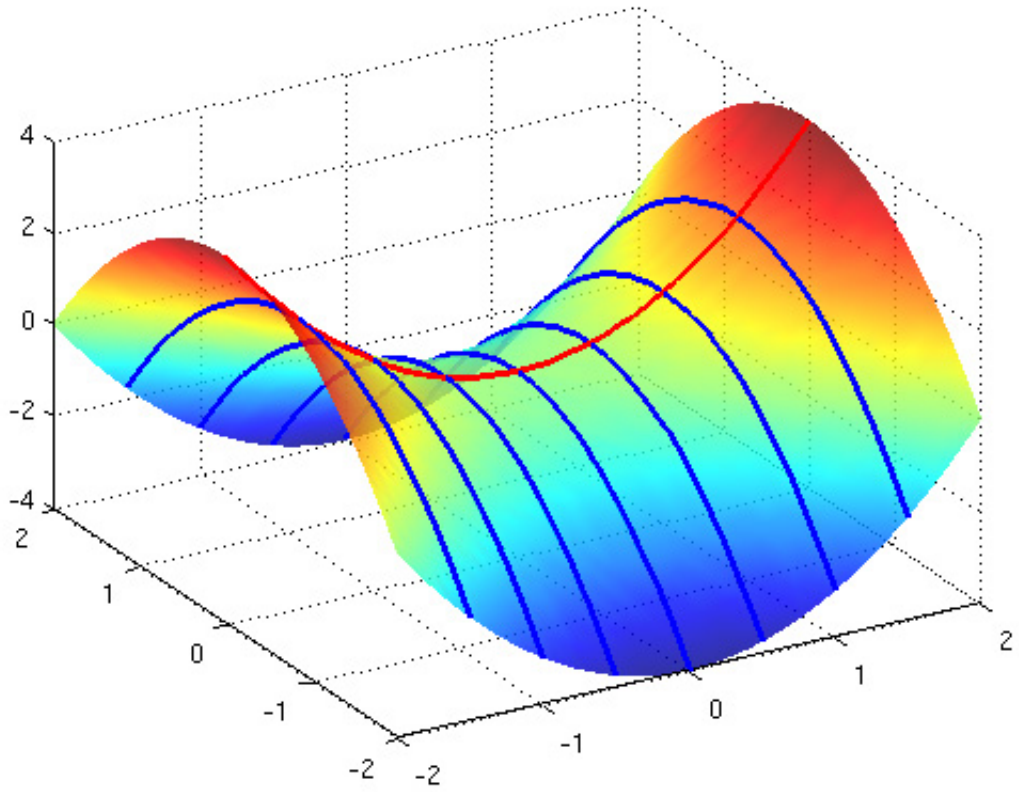


FIGURE 7





minmax1.jpg

Figure 8