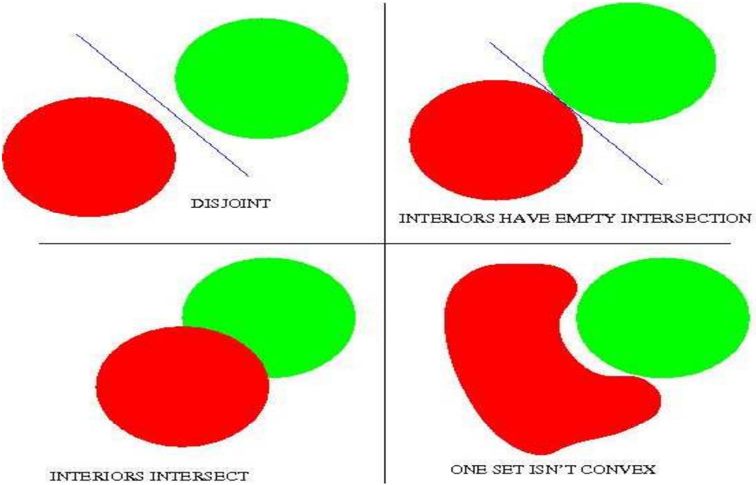
Outline

Roadmap for the NPP segment:

1. Preliminaries: role of convexity
2. Existence of a solution
3. Necessary conditions for a solution: inequality constraints
4. The constraint qualification
5. The Lagrangian approach
6. Interpretation of the Lagrange Multipliers
7. Constraints that are binding vs satisfied with equality
8. Necessary conditions for a solution: eq and ineq constraints
9. Sufficiency conditions
    1. Bordered Hessians
    2. Pseudo-concavity
        1. The relationship between quasi- and pseudo-concavity
10. The basic sufficiency theorem

Separating Hyperplanes

Theorem: If A and B are convex, with $\text{int}(A) \cap \text{int}(B) = \emptyset$, then A and B can be separated by a hyperplane.



Want to be able to separate by a hyperplane

- upper contour set of objective function
- constraint set

Role of quasi-ness properties of functions

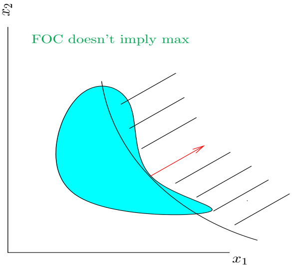
- A function is *quasi-concave* (*quasi-convex*) if all of its upper (lower) contour sets are convex
- A function is *strictly quasi-concave (convex)* if
  - all of its upper (lower) contour sets are "strictly" convex sets (no flat edges)
  - all of its level sets have empty interior

To make everything work nicely

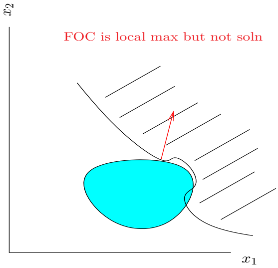
1. Require your objective function to be strictly quasi-concave
2. Define your feasible set as the intersection of lower contour sets of quasi-convex *functions*,
   - e.g., $g^j(x) \leq b_j$, for $j = 1, ...m$.
   - intersection of convex sets is convex
   - your feasible set is convex

If there's only one constraint:

- First order necessary condition is "kissing point of mutual tangency"
  - gradients of objective and constraint are collinear
- appropriate quasiness assumptions ensure that FOC is also sufficient



Constraint set is not convex



Upper contour set of objective not convex

Existence

Preliminaries:

- The canonical form of the nonlinear programming problem (NPP)

$$\text{maximize } f(\mathbf{x}) \text{ subject to } \mathbf{g}(\mathbf{x}) \leq \mathbf{b} \tag{1}$$

- Conditions for existence of a solution the NPP:
  Theorem: If $f : A \to \mathbb{R}$ is continuous and *strictly quasi-concave*, and A is compact, nonempty and convex, then f attains a unique maximum on A.
- Want to apply this theorem to guarantee a solution to (1)
  - $\{x : \mathbf{g}(\mathbf{x}) \leq \mathbf{b}\}$ convex if each g^j is quasi-convex
  - but quasi-ness can't guarantee us compactness
- Theorem: If $f : \mathbb{R}^n \to \mathbb{R}$ is continuous and strictly quasi-concave and $g^j : \mathbb{R}^n \to \mathbb{R}$ is *quasi-convex* for each j, then *if* f attains a local maximum on $A = \{\mathbf{x} \in \mathbb{R}^n : \forall j, g^j(\mathbf{x}) \leq b_j\}$, this max is the unique global maximum of f on A.

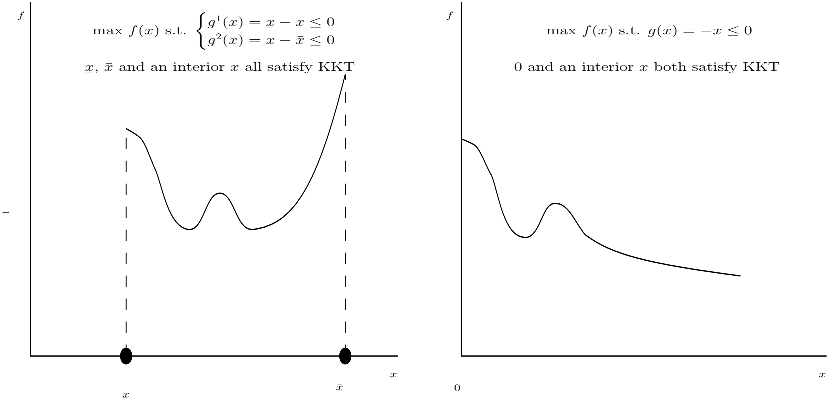
KKT necessary conditions for a soln: inequality constraints

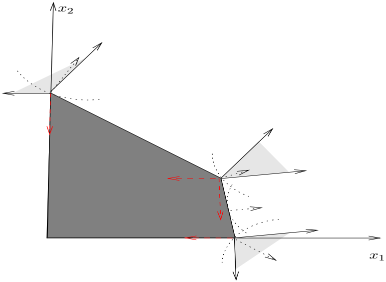
- The KKT conditions in Mantra format:
  (Except for a bizarre exception) a necessary condition for **x** to solve a constrained maximization problem is that the gradient vector of the objective function at **x** belongs to the nonnegative cone defined by the gradient vectors of the constraints that are satisfied with equality at **x**.
- The KKT conditions in math format: Note absence of Lagrangian!!
  If $\bar{\mathbf{x}}$ solves the maximization problem *and the constraint qualification holds* at $\bar{\mathbf{x}}$ then there exists a vector $\bar{\boldsymbol{\lambda}} \in \mathbb{R}^m_+$ s.t.

  $$\bigtriangledown f(\bar{\mathbf{x}})' = \bar{\boldsymbol{\lambda}}' J\mathbf{g}(\bar{\mathbf{x}})$$

  Moreover, $\bar{\boldsymbol{\lambda}}$ has the property: $g^j(\bar{\mathbf{x}}) < b_j \implies \bar{\lambda}_j = 0$.
- The KKT conditions vs the Lagrangian: exactly the same.
  - The Lagrangian is scalar-based, “long-hand”
  - The KKT is in vector form

Complementary Slackness conditions: univariate function





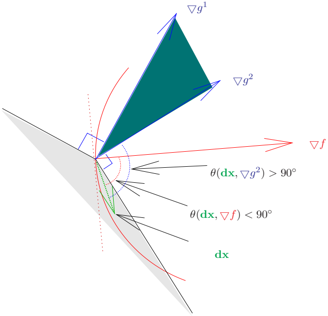
a necessary condition ... is that the gradient vector of the objective function at *x* belongs to the nonnegative cone defined by the gradient vectors of the constraints satisfied with equality at *x*

- 3 dotted curves represent level sets of 3 different objective functions
- in each case, dashed black arrows (gradients of objectives) belong to appropriate non-negative cone, defined by solid arrows
- red arrows: $\bigtriangledown$'s of constraints not satisfied with $=$ at point being examined

Necessary conditions for a Maximum

In this picture the mantra fails

- $\bigtriangledown f$ doesn't belong to cone defined by $\bigtriangledown g^1$ and $\bigtriangledown g^2$
- dashed line perp^ular to $\bigtriangledown f$ intersects the interior of constraint set
  - angle between **dx** and $\bigtriangledown f$ is acute
  - angles between **dx** and the $\bigtriangledown g^i$'s are obtuse
- direction **dx** increases f & strictly decreases BOTH constraints
- conclude: the x value at tail of red arrow can't solve the NPP

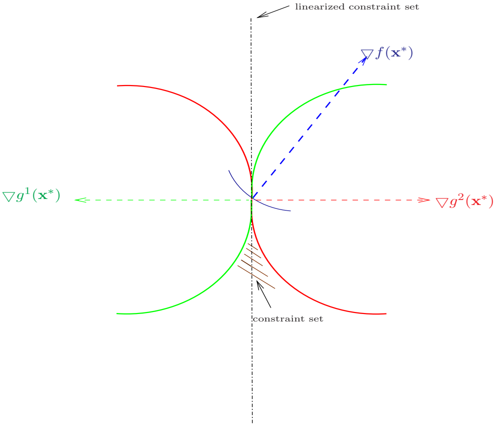


KTGradients

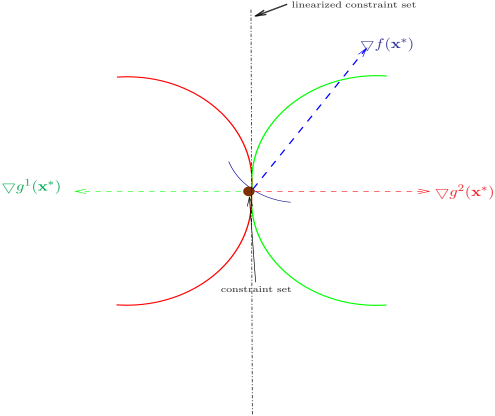
Important issues

- The role of the constraint qualification (CQ):
  1. it ensures that the linearized version of the constraint set is, locally, a good approximation of the true nonlinear constraint set.
  2. a sufficient (but not necessary) condition for the CQ to hold
     - the CQ will be satisfied at x if the gradients of the constraints that are satisfied with equality at x form a linear independent set
- Interpretation of the Lagrangian
- Constraint satisfied with equality vs binding constraints.
- The KKT and equality constraints

The Constraint Qualification

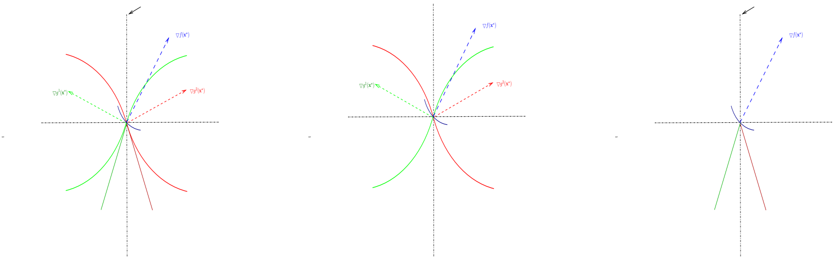


The Constraint Qualification with quasi-convex constraints



KKT conditions are necessary for soln to problem in the right-hand panel

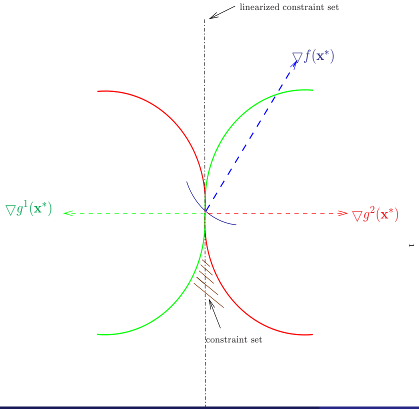
- they are necessary for soln to problem in the middle panel
  - ◇ only if the two problems are essentially the same in nbd of soln to former

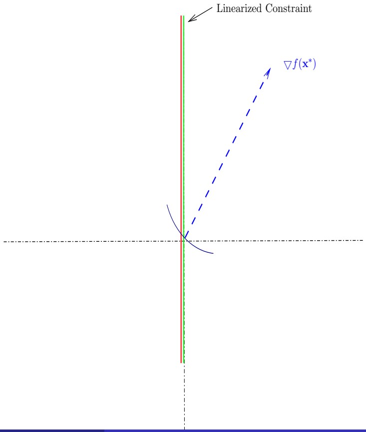


CQ fails if linearized constraint set differs from original set

KKT conditions are necessary for soln to problem in the right-hand panel

- since it doesn't have a solution, the KKT conditions can't be satisfied
- KKT conditions tell us nothing about the problem in the left panel





KKT and the Lagrangian Approach

The Lagrangian function for $f : \mathbb{R}^n \to \mathbb{R}$, $g : \mathbb{R}^n \to \mathbb{R}^m$ and $b \in \mathbb{R}^m$.

$$L(x, \boldsymbol{\lambda}) \quad = \quad f(x) + \boldsymbol{\lambda}(b - g(x)) \quad = \quad f(x) + \sum_{j=1}^{m} \lambda_j (b_j - g^j(x))$$

The first order conditions for an extremum of L on $\mathbb{R}^n \times \mathbb{R}^m_+$ are: $\exists (\bar{x}, \bar{\boldsymbol{\lambda}})$ s.t.

$$\begin{aligned} \text{for } i = 1, \ldots n, \partial L(\bar{x}, \bar{\boldsymbol{\lambda}})/\partial x_i &= 0 && (2) \\ \text{for } j = 1, \ldots m, \partial L(\bar{x}, \bar{\boldsymbol{\lambda}})/\partial \lambda_j &\geq 0 && (3) \\ \bar{\lambda}_j \partial L(\bar{x}, \bar{\boldsymbol{\lambda}})/\partial \lambda_j &= 0. && (4) \end{aligned}$$

Compare KKT with (2)

$$\begin{aligned} \bigtriangledown f(\bar{\mathbf{x}})^T &= \bar{\boldsymbol{\lambda}}^T J\mathbf{g}(\bar{\mathbf{x}}) \\ \partial f(\bar{x})/\partial x_i &= \sum_{j=1}^{m} \bar{\lambda}_j \cdot \partial g^j(\bar{x})/\partial x_i \end{aligned}$$

(3) says for $j = 1, \ldots m$, $(b_j - g^j(\bar{x})) \geq 0$ or, in vector notation, $(b - g(\bar{x})) \geq 0$.
(4) "*complementary slackness*": $(b_j - g^j(\bar{x})) > 0$ implies $\bar{\lambda}_j = 0$;
Note no nonnegativity constraints: we treat these like any other constraints

Lagrangians as "shadow values"

Theorem: At a solution $(\bar{x}, \bar{\boldsymbol{\lambda}})$ to the NPP, $L(\bar{x}, \bar{\boldsymbol{\lambda}}) = f(\bar{x})$.

$$\underbrace{M(b)}_{\text{value function}} \quad = \quad L(\bar{x}(b), \bar{\boldsymbol{\lambda}}(b)) \quad = \quad f(\bar{x}) + \underbrace{\bar{\boldsymbol{\lambda}}(b - g(\bar{x}))}_{=0} \quad = \quad f(\bar{x}(b)).$$

Interpretation: $\bar{\lambda}^j$ is the "shadow value" of the j'th constraint:

$$\frac{dM(b)}{db_j} = \frac{dL(\bar{x}(b), \bar{\boldsymbol{\lambda}}(b))}{db_j} \quad = \quad \sum_{i=1}^{n} \left\{ f_i(\bar{x}(b)) - \sum_{k=1}^{m} \bar{\lambda}_k(b) g_i^k(\bar{x}(b)) \right\} \frac{dx_i}{db_j}$$
$$+ \sum_{k=1}^{m} \frac{d\lambda_k}{db_j} (b_k - g^k(\bar{x}(b))) + \bar{\lambda}_j(b)$$

- For each i, the term in curly brackets is zero, by the KT conditions.
- For each k,
  - If $(b_k - g^k(\bar{x}(b))) = 0$, then $\frac{d\lambda_k}{db_j}(b_k - g^k(\bar{x}(b)))$ is zero.
  - If $(b_k - g^k(\bar{x}(b))) < 0$, then $\bar{\lambda}_k(\cdot) = 0$ on a nbd of b, so $\frac{d\lambda_k(\cdot)}{db_k} = 0$.
- The only term remaining is $\bar{\lambda}_j(b)$.
- Conclude that $\frac{dM(b)}{db_j} = \bar{\lambda}_j(b)$

Interpretation of the Lagrangian: One inequality constraint

Gradient of objective and constraint are collinear: $\forall i, \frac{\partial f(x^*)}{\partial x_i} = \lambda^* \frac{\partial g(x^*)}{\partial x_i}$.

- i.e., λ is the ratio of the norms of the two gradients
- in both figures, dashed line represents relaxation of constraint by $\Delta b = 1$.
- the arrows represent $\bigtriangledown f(x)$ and $\bigtriangledown g(x)$.
    - in one panel $||\bigtriangledown f(x)||$ small; $||\bigtriangledown g(x)||$ big;
    - in the other, $||\bigtriangledown f(x)||$ big; $||\bigtriangledown g(x)||$ small

  which is which? Picture provides no info about $\bigtriangledown f$, only about $\bigtriangledown g$.

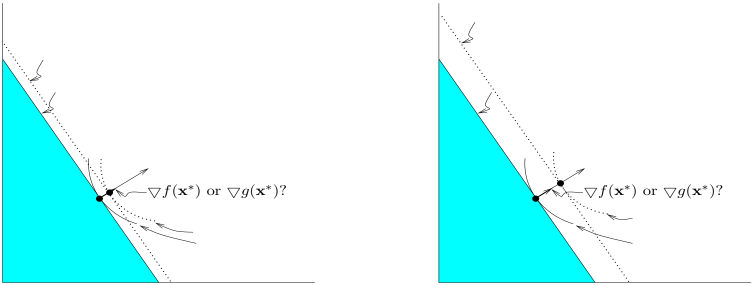
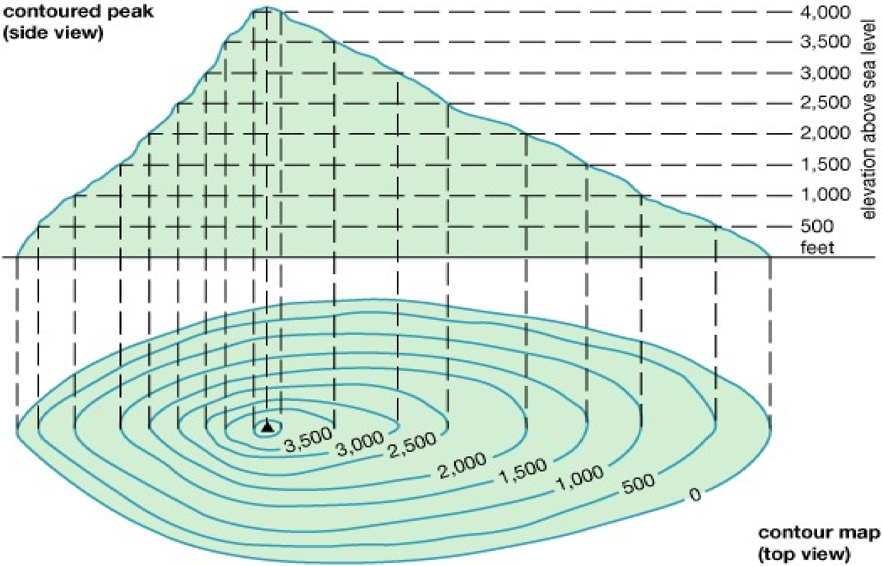


FIGURE 1. Interpretation of the Lagrangian


contoured peak
(side view)
4,000
3,500
3,000
2,500
2,000
1,500
1,000
500
feet
elevation above sea level
3,500
3,000
2,500
2,000
1,500
1,000
500
0
contour map
(top view)

Gradient of objective and constraint are collinear

- λ is the ratio of the norms of the two gradients
  - left panel: $|| \bigtriangledown f(x)||$ small; $|| \bigtriangledown g(x)||$ big; interpretation
  - right panel: $|| \bigtriangledown f(x)||$ big; $|| \bigtriangledown g(x)||$ small; interpretation
- λ is the “shadow value” of the constraint: “bang for a buck”

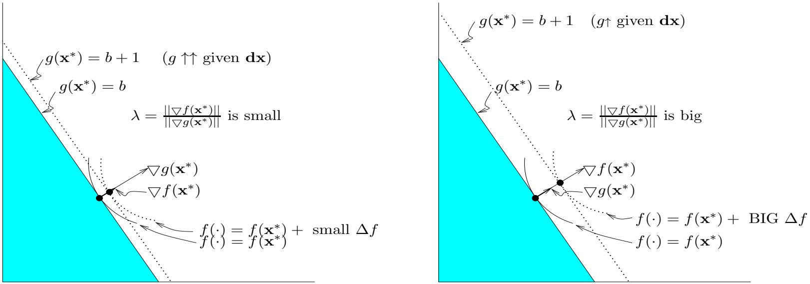
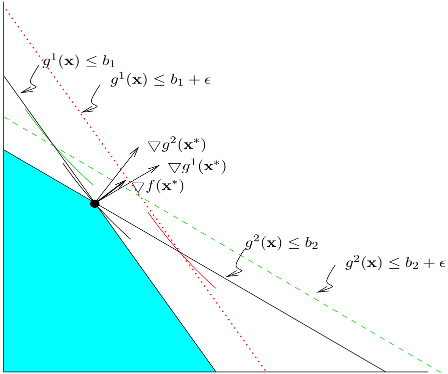


Figure 1. Interpretation of the Lagrangian

Interpretation of the Lagrangian: Two inequality constraints

Gradient of objective and red constraint are nearly collinear

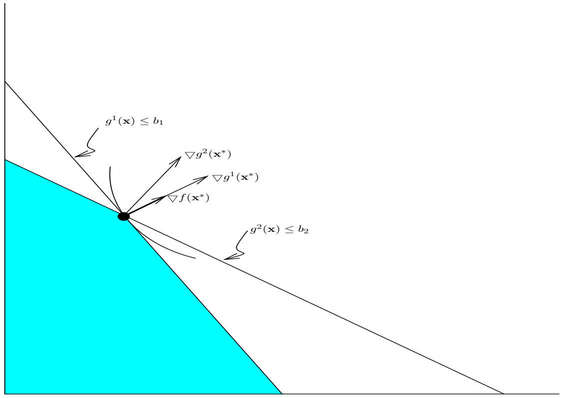
- λ^1 is relatively large
  - relatively large gain utility gain from $\Delta b^1 = \varepsilon$.
- λ^2 is relatively small
  - relatively small utility gain gain from $\Delta b^2 = \varepsilon$.
- In limit, when $\bigtriangledown f$ is collinear with $\bigtriangledown g^1$, adding ε to b^2 changes *nothing*



Constraint satisfied with equality vs binding

Informal defn: a constraint is *binding* if the maximized value of objective increases when the constraint is slightly relaxed.

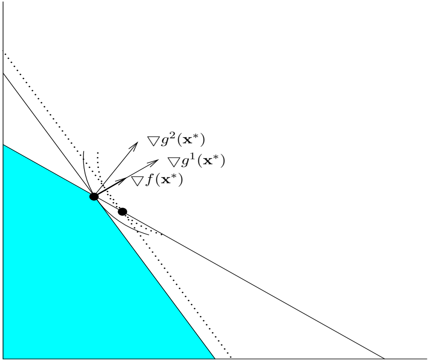
- which constraint is binding in the figure below?

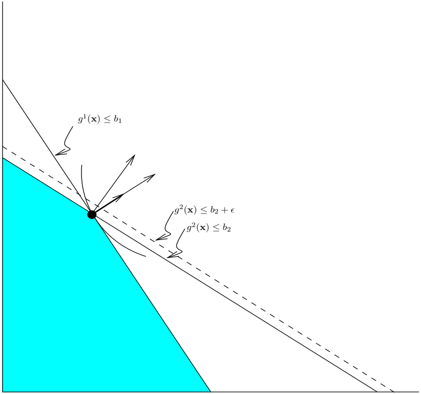


One constraint binding; the other is satisfied with equality

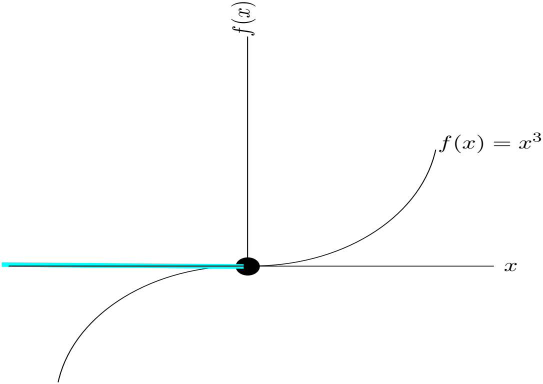
$g^j(x) = b_j$, $j = 1, 2$ i.e., both constraints satisfied with equality

- when b_1 increases, optimum moves, utility increases, so g^1 is binding
- when b_2 increases, optimum doesn't move, so not g^2 is *not* binding





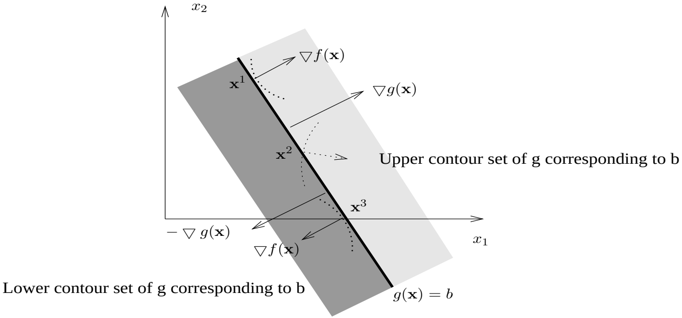
$\lambda > 0 \Longrightarrow$ constraint binding but bindingness $\not\Longrightarrow \lambda > 0$



KKT with equality constraints defined as inequalities

KKT with equality constraints

- $g(x) \leq b$ vs $g(x) = b$.
- Two differences: in the latter case but not the former
  - the constraint can bind “in either direction”
  - there is no complementary slackness condition



KKT necessary conditions: equality & inequality constraints

The KKT conditions in full generality

- $f : \mathbb{R}^n \to \mathbb{R}$, $g : \mathbb{R}^n \to \mathbb{R}^m$, $h : \mathbb{R}^n \to \mathbb{R}^\ell$ are twice continuously diff$^{\text{able}}$
- $b \in \mathbb{R}^m$ and $c \in \mathbb{R}^\ell$.
- The canonical form of the nonlinear programming problem (NPP)

$$\text{maximize } f(\mathbf{x}) \text{ subject to } \mathbf{g}(\mathbf{x}) \leq \mathbf{b} \text{ and } \mathbf{h}(\mathbf{x}) = \mathbf{c}$$

- The KKT conditions:
  If $\bar{\mathbf{x}}$ solves the maximization problem *and the constraint qualification holds* at $\bar{\mathbf{x}}$ then there exist vectors $\bar{\boldsymbol{\lambda}} \in \mathbb{R}^m_+$ and $\bar{\boldsymbol{\mu}} \in \mathbb{R}^\ell$ s.t.

$$\bigtriangledown f(\bar{\mathbf{x}})' = \bar{\boldsymbol{\lambda}}' J\mathbf{g}(\bar{\mathbf{x}}) + \bar{\boldsymbol{\mu}}' J\mathbf{h}(\bar{\mathbf{x}})$$

  Moreover, $\bar{\boldsymbol{\lambda}}$ has the property: $g^j(\bar{\mathbf{x}}) < b_j \Longrightarrow \bar{\lambda}_j = 0$.
- Note the two differences between $\bar{\boldsymbol{\lambda}}$ and $\bar{\boldsymbol{\mu}}$
  - $\bar{\boldsymbol{\lambda}}$ must be nonnegative;
  - there is no complementary slackness condition on $\bar{\boldsymbol{\mu}}$

KKT necessary conditions: an example

This is question 2 on the NPP1 problem set, I'm going to give you a start on it

$$
\max_{x_1,x_2} \quad 2x_1x_2 + 9x_2 - 2x_1^2 - 2x_2^2 \quad \text{s.t.}
$$

$$
\begin{aligned}
g_1: \quad & -x_1 & \leq & \quad 0 \\
g_2: \quad & -x_2 & \leq & \quad 0 \\
g_3: \quad & 4x_1 + 3x_2 & \leq & \quad 10 \\
g_4: \quad & 4x_1^2 - x_2 & \leq & \quad 2
\end{aligned}
$$

KKT necessary conditions

$$
\triangledown f(x)' \quad = \quad \boldsymbol{\lambda}' Jg(x)
$$

$$
\begin{bmatrix} 2x_2 - 4x_1, & 2x_1 - 4x_2 + 9 \end{bmatrix} \quad = \quad \begin{bmatrix} \lambda_1, & \lambda_2, & \lambda_3, & \lambda_4 \end{bmatrix} \begin{bmatrix} -1, & 0 \\ 0, & -1 \\ 4, & 3 \\ 8x_1, & -1 \end{bmatrix}
$$

KKT example: check if $x = 0$ satisfies necessary conditions

KKT Conditions:

$$\begin{bmatrix} 2x_2 - 4x_1, & 2x_1 - 4x_2 + 9 \end{bmatrix} = \begin{bmatrix} \lambda_1, & \lambda_2, & \lambda_3, & \lambda_4 \end{bmatrix} \begin{bmatrix} -1, & 0 \\ 0, & -1 \\ 4, & 3 \\ 8x_1, & -1 \end{bmatrix}$$

If $x = 0$, then $g_3(x) = 4x_1 + 3x_2 = 0 < 10$ and $g_4(x) = 4x_1^2 - x_2 = 0 < 2$.

Complementary slackness condition now implies $\lambda_3 = \lambda_4 = 0$.

KKT conditions now become :

$$\begin{bmatrix} 0, & 9 \end{bmatrix} = \begin{bmatrix} \lambda_1, & \lambda_2 \end{bmatrix} \begin{bmatrix} -1, & 0 \\ 0, & -1 \end{bmatrix} = \begin{bmatrix} -\lambda_1, & -\lambda_2 \end{bmatrix}$$

Can't solve 2nd equation if $\boldsymbol{\lambda} \geq 0$; conclude $x = 0$ can't solve NPP

KKT example: check if $x_1 > x_2 = 0$ satisfies KKT condition

KKT Conditions:

$$\begin{bmatrix} 2x_2 - 4x_1, & 2x_1 - 4x_2 + 9 \end{bmatrix} = \begin{bmatrix} \lambda_1, & \lambda_2, & \lambda_3, & \lambda_4 \end{bmatrix} \begin{bmatrix} -1, & 0 \\ 0, & -1 \\ 4, & 3 \\ 8x_1, & -1 \end{bmatrix}$$

If $x_2 = 0$, then $g_4(x) = 4x_1^2 - x_2 \leq 2 \implies x_1 \leq \sqrt{1/2}$

$$x_1 \leq \sqrt{1/2} \implies g_3(x) = 4x_1 + 3x_2 \leq 4\sqrt{1/2} < 10.$$

If $x_1 > x_2 = 0$ and $g_4(x) \leq 2$, then $g_1(x) < 0$ and $g_3(x) < 0$.

Complementary slackness condition now implies $\lambda_1 = \lambda_3 = 0$.

KKT conditions now become :

$$\begin{bmatrix} -4x_1, & 2x_1 + 9 \end{bmatrix} = \begin{bmatrix} \lambda_2, & \lambda_4 \end{bmatrix} \begin{bmatrix} 0, & -1 \\ 8x_1, & -1 \end{bmatrix} = \begin{bmatrix} 8\lambda_4 x_1, & -\lambda_2 - \lambda_4 \end{bmatrix}$$

Can't solve either equation if $\boldsymbol{\lambda} \geq 0$; conclude $x_1 > x_2 = 0$ can't solve NPP