EE/Econ 458
Introduction to Optimization

J. McCalley

Electricity markets and tools

Day-ahead → SCUC and SCED

Real-time → SCED

BOTH LOOK LIKE THIS

SCUC:
$\underline{x}$ contains discrete & continuous variables.

Minimize $f(\underline{x})$

subject to

$\underline{h}(\underline{x})=\underline{c}$

$\underline{g}(\underline{x})\underline{\leq}\ \underline{b}$

SCED:
$\underline{x}$ contains only continuous variables.

Optimization Terminology

An optimization problem or a mathematical program or a mathematical programming problem.

$$\text{Minimize } f(\underline{x})$$
$$\text{subject to}$$
$$\underline{h}(\underline{x})=\underline{c}$$
$$\underline{g}(\underline{x})\underline{\geq}\ \underline{b}$$

$f(\underline{x})$: Objective function

$\underline{x}$: Decision variables

$\underline{h}(\underline{x})=\underline{c}$: Equality constraint

$\underline{g}(\underline{x})\underline{\geq}\ \underline{b}$: Inequality constraint

$\underline{x}^*$: solution

Classification of Optimization Problems

Continuous Optimization

Unconstrained Optimization
Bound Constrained Optimization
Derivative-Free Optimization
Global Optimization
Linear Programming
Network Flow Problems
Nondifferentiable Optimization
Nonlinear Programming
Optimization of Dynamic Systems
Quadratic Constrained Quadratic Programming
Quadratic Programming
Second Order Cone Programming
Semidefinite Programming
Semiinfinite Programming

Discrete and Integer Optimization

Combinatorial Optimization
    Traveling Salesman Problem
Integer Programming
    Mixed Integer Linear Programming
    Mixed Integer Nonlinear Programming

Optimization Under Uncertainty

Robust Optimization
Stochastic Programming
Simulation/Noisy Optimization
Stochastic Algorithms

Complementarity Constraints and Variational Inequalities

Complementarity Constraints
Game Theory
Linear Complementarity Problems
Mathematical Programs with Complementarity Constraints
Nonlinear Complementarity Problems

Systems of Equations

Data Fitting/Robust Estimation
Nonlinear Equations
Nonlinear Least Squares

Systems of Inequalities

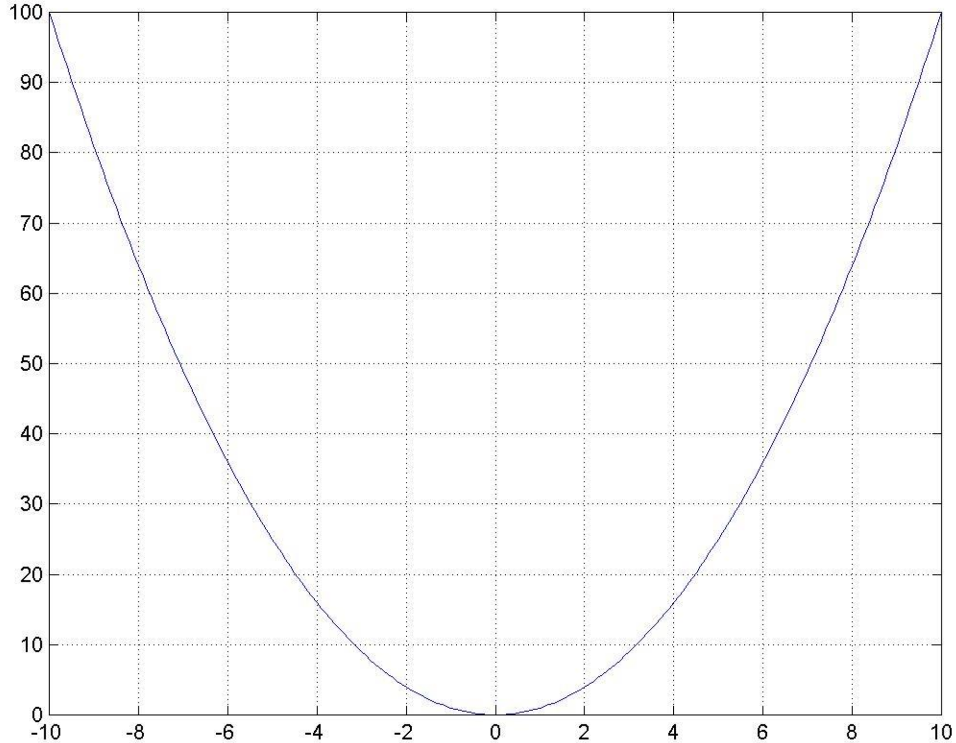
Multiobjective Optimization

http://www.neos-guide.org/NEOS/index.php/Optimization_Tree

Convex functions

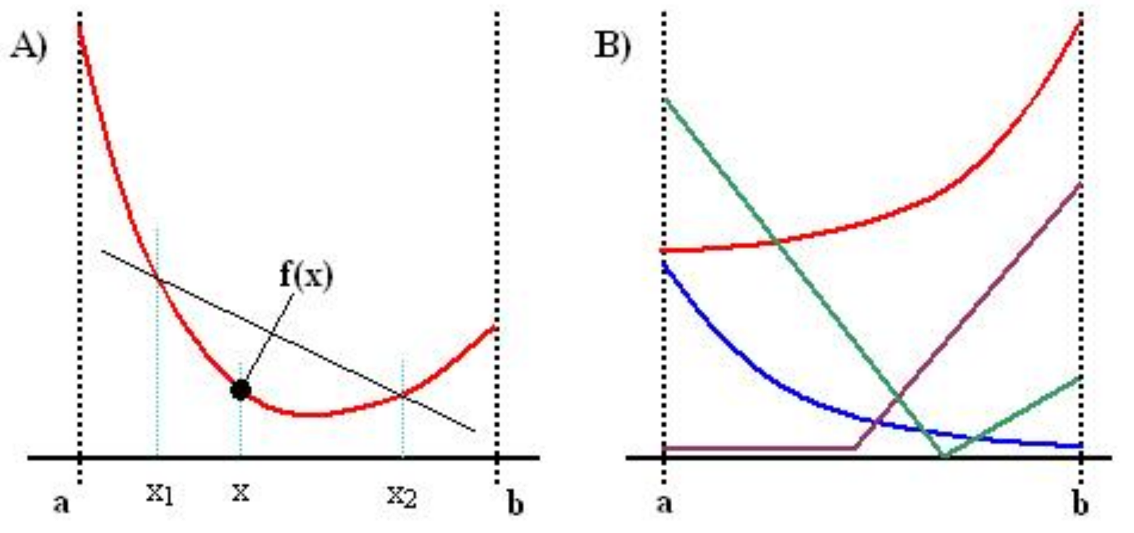
Definition #1: A function f(x) is convex in an interval if its second derivative is positive on that interval.

Example: $f(x)=x^2$ is convex since $f'(x)=2x$, $f''(x)=2>0$



Convex functions

The second derivative test is sufficient but not necessary.



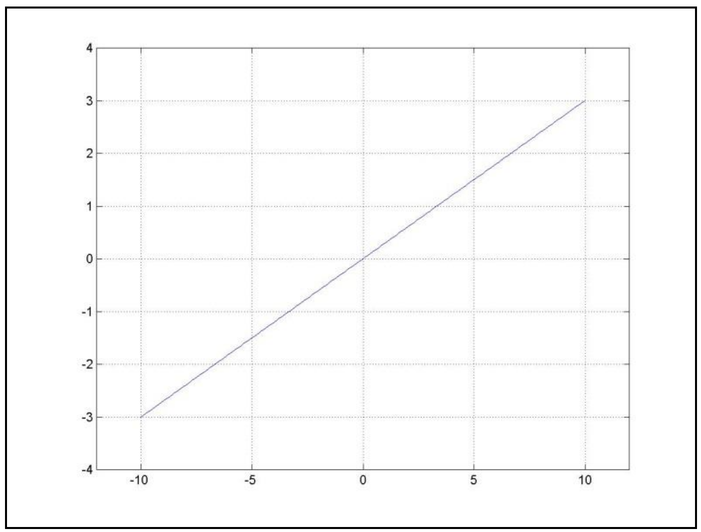
<u>Definition #2</u>: A function f(x) is convex if a line drawn between any two points on the function remains on or above the function in the interval between the two points.

www.ebyte.it/library/docs/math09/AConvexInequality.html

Convex functions

<u>Definition #2</u>: A function f(x) is convex if a line drawn between any two points on the function remains on or above the function in the interval between the two points.

Is a linear function convex?

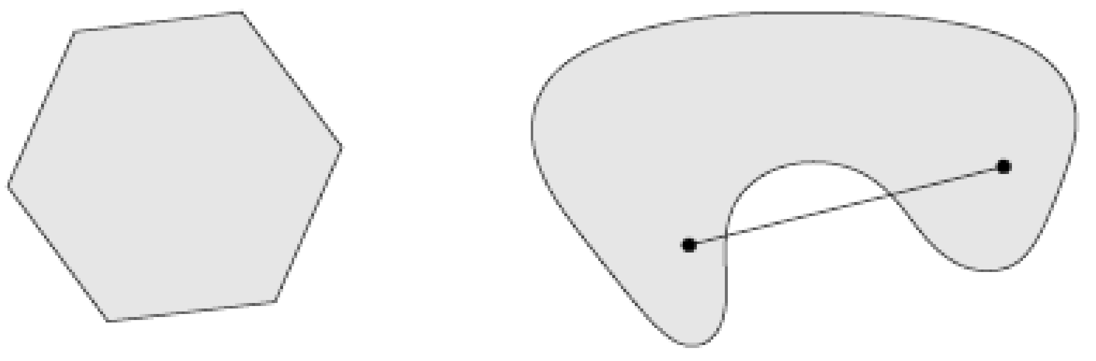


Answer is "yes" since a line drawn between any two points on the function remains ***<u>on</u>*** the function.

Convex Sets

Definition #3: A set C is convex if a line segment between any two points in C lies in C.

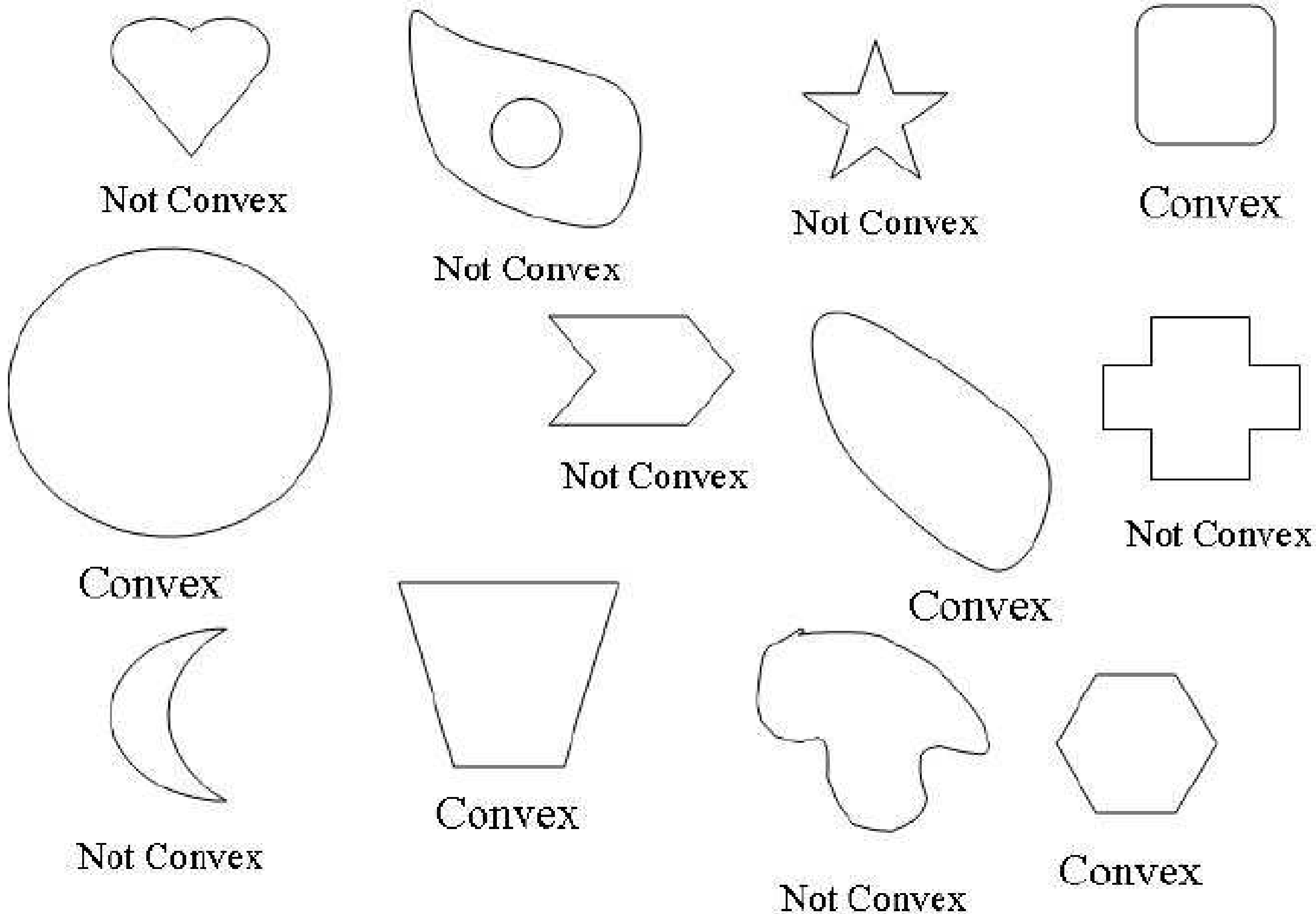
Ex: Which of the below are convex sets?



The set on the left is convex. The set on the right is not.

Convex Sets

Definition #3: A set C is convex if a line segment between any two points in C lies in C.



S. Boyd and L. Vandenberghe, “Convex optimization,” Cambridge University Press, 2004.

Global vs. local optima

Example: Solve the following:

Minimize $f(x)=x^2$

Solution: $f'(x)=2x=0 \Rightarrow x^*=0$.

This solution is a local optimum.

It is also the global optimum.

Example: Solve the following:

Minimize $f(x)=x^3-17x^2+80x-100$

Solution: $f'(x)=3x^2-34x+80=0$

Solving the above results in x=3.33 and x=8.

Issue#1: Which is the best solution?

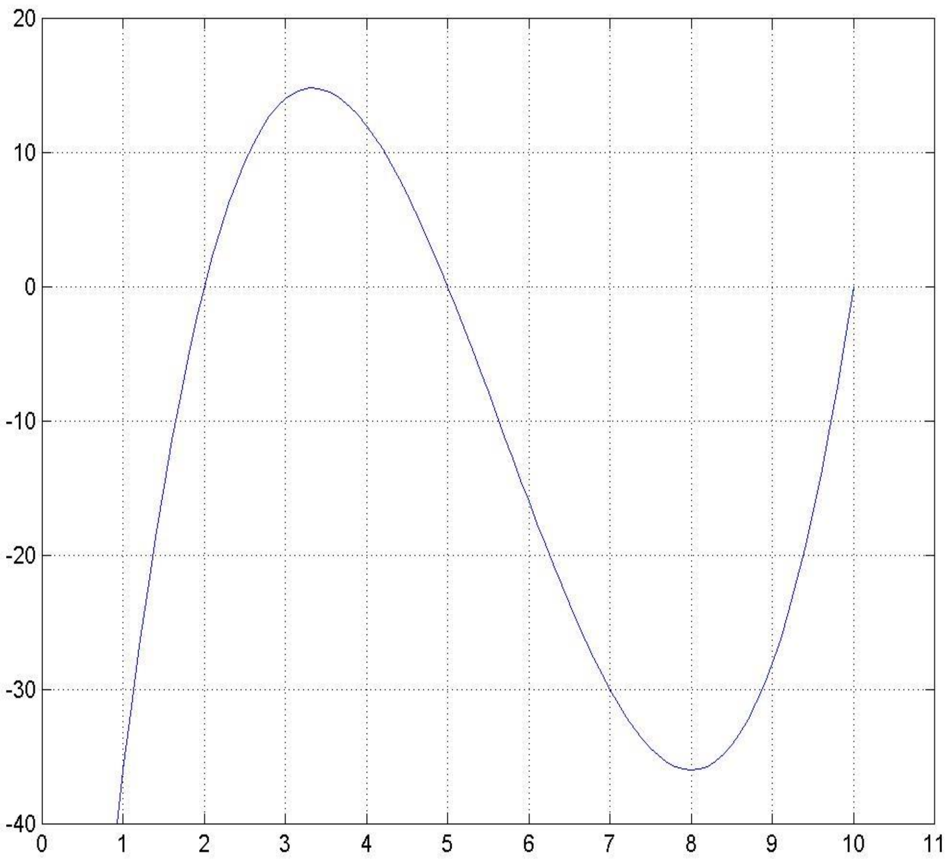
Issue#2: Is the best solution the global solution?

Global vs. local optima

<u>Example</u>: Solve the following:

Minimize $f(x)=x^3-17x^2+80x-100$

Solution: $f'(x)=3x^2-34x+80=0$. Solving results in $x=3.33$, $x=8$.



Issue#1: Which is the best solution?

➔ $x=8$

Issue#2: Is the best solution the global solution?

➔ No! It is unbounded.

Convexity & global vs. local optima

➔When minimizing a function, if we want to be sure that we can get a global solution via differentiation, we need to impose some requirements on our objective function.

➔We will also need to impose some requirements on the feasible set S (set of possible values the solution $\underline{x}^*$ may take).

$$\text{Min } f(\underline{x}) \quad\quad\quad \min f(\underline{x})$$

$$\text{subject to} \quad\quad\quad \text{subject to}$$

$$\left.\begin{array}{l} \underline{h}(\underline{x})=\underline{c} \\ \underline{g}(\underline{x})\underline{\geq}\ \underline{b} \end{array}\right. \xrightarrow{\text{Feasible set}} \underline{x} \in S$$

<u>Definition</u>: If f($\underline{x}$) is a convex function, and if S is a convex set, then the above problem is a *convex programming problem*.

<u>Definition</u>: If f($\underline{x}$) is not a convex function, or if S is not a convex set, then the above problem is a *non-convex programming problem*.

Convex vs. nonconvex programming problems

The desirable quality of a convex programming problem is that any *locally optimal solution* is also a *globally optimal solution*. ➔If we have a method of finding a locally optimal solution, that method also finds for us the globally optimum solution.

The undesirable quality of a non-convex programming problem is that any method which finds a locally optimal solution does not necessarily find the globally optimum solution.

MATHEMATICAL PROGRAMMING

- **Convex**

  We address convex programming problems in addressing linear programming.

- **Non-convex**

  We will also, later, address a special form of non-convex programming problems called integer programs.

A convex programming problem

Two variables with one equality-constraint

$$\min \; f(x_1, x_2)$$
$$s.t. \;\; h(x_1, x_2) = c$$

We focus on this one, but conclusions we derive will also apply to the other two. The benefit of focusing on this one is that we can visualize it.

Multi-variable with one equality-constraint.

$$\min \; f(\underline{x})$$
$$\text{s.t.} \;\; h(\underline{x}) = c$$

Multi-variable with multiple equality-constraints.

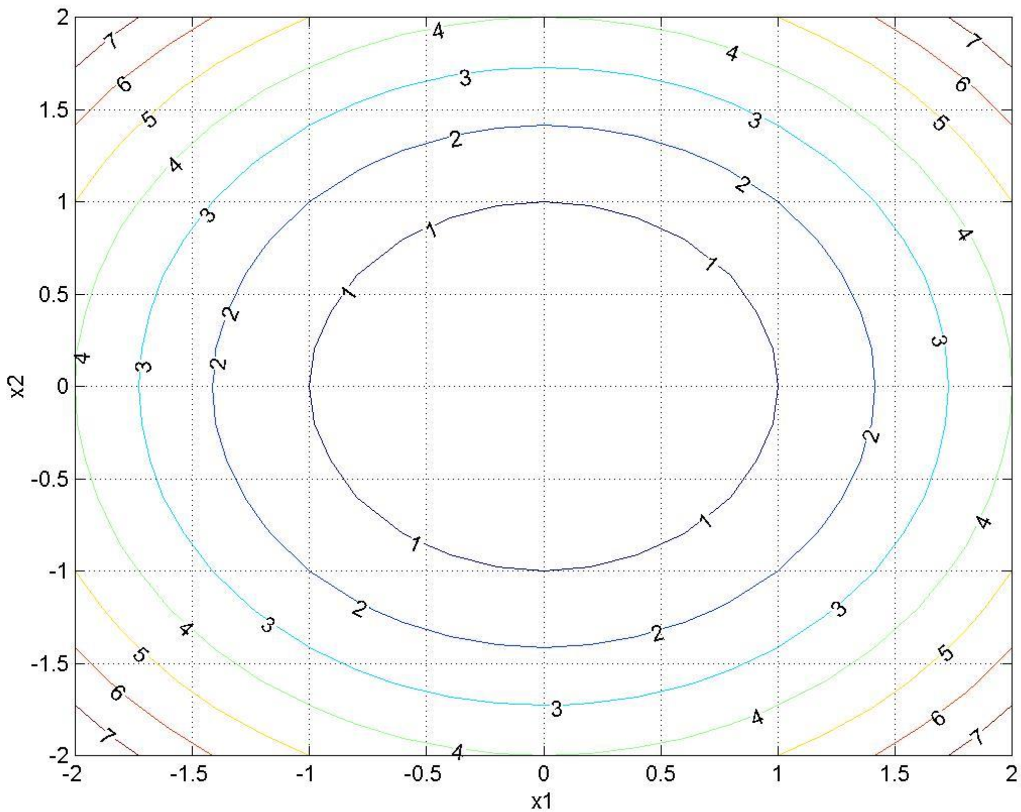
$$\min \; f(\underline{x})$$
$$\text{s.t.} \;\; \underline{h}(\underline{x}) = \underline{c}$$

Contour maps

- Definition: A contour map is a 2-dimensional plane, i.e., a coordinate system in 2 variables, say, x_1, x_2, that illustrates curves (contours) of constant functional value $f(x_1, x_2)$.

Example: Draw the contour map for $f(x_1, x_2) = x_1^2 + x_2^2$

```
[X,Y] = meshgrid(-
2.0:.2:2.0,-2.0:.2:2.0);
Z = X.^2+Y.^2;
[c,h]=contour(X,Y,Z);
clabel(c,h);
grid;
xlabel('x1');
ylabel('x2');
```



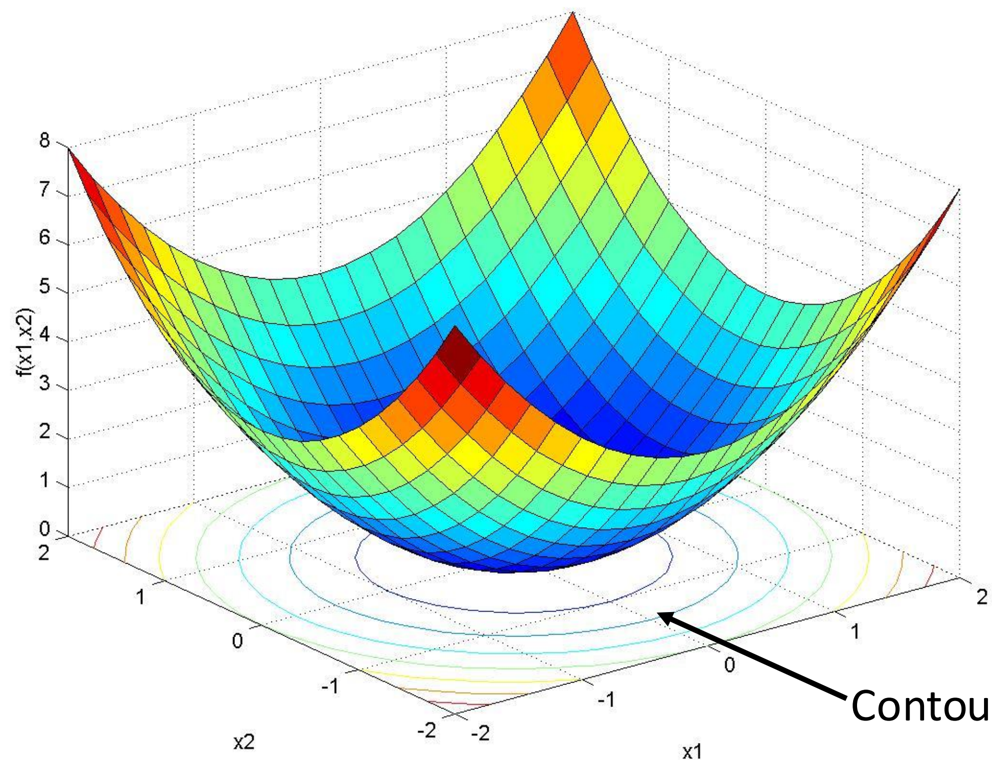
Contour maps and 3-D illustrations

.

Example: Draw the 3-D surface for $f(x_1, x_2) = x_1^2 + x_2^2$

```
[X,Y] = meshgrid(-
2.0:.2:2.0,-2.0:.2:2.0);
Z = X.^2+Y.^2;
surfc(X,Y,Z)
xlabel('x1')
ylabel('x2')
zlabel('f(x1,x2)')
```

Height is f(x)



Contours

Each contour of fixed value *f* is the projection onto the *x1-x2* plane of a horizontal slice made of the 3-D figure at a value *f* above the *x1-x2* plane.

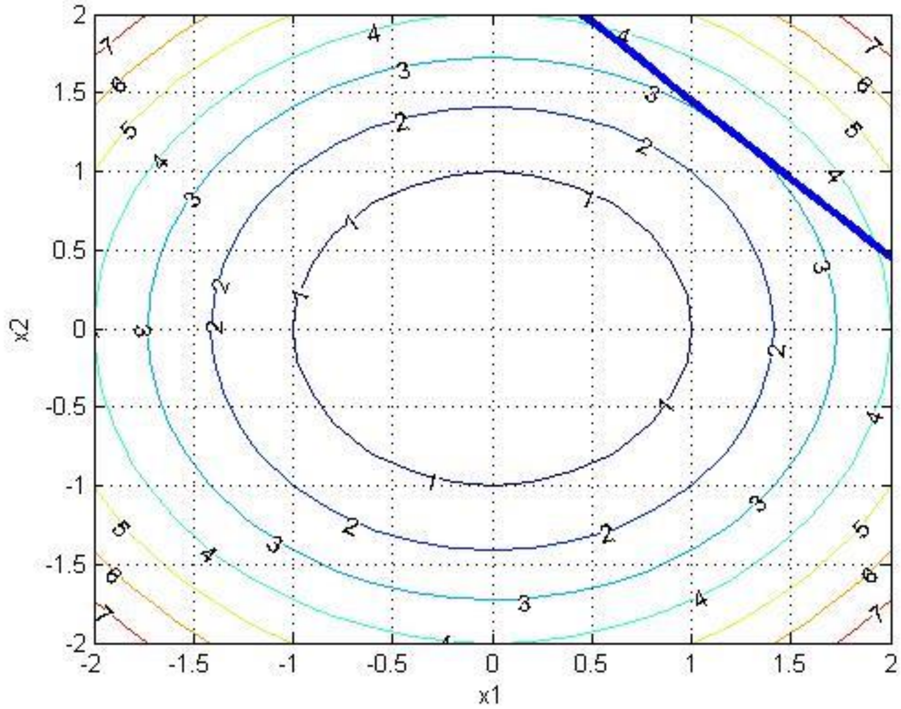
Solving a convex program: graphical analysis

<u>Example</u>: Solve this convex program: $\min\ f(x_1, x_2) = x_1^2 + x_2^2$

$$s.t.\quad h(x_1,x_2) = x_1 + x_2 = \sqrt{6}$$

A straight line is a convex set because a line segment between any two points on it remain on it.

$$x_1 + x_2 = \sqrt{6} \Rightarrow x_2 = -x_1 + \sqrt{6}$$



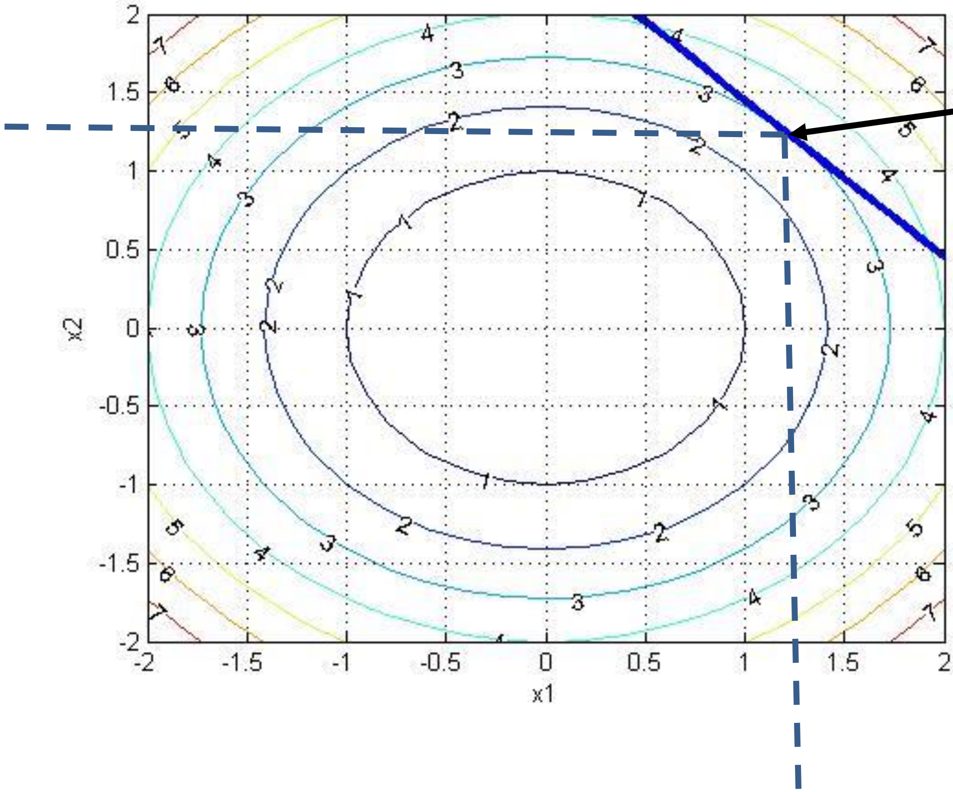
Superimpose this relation on top of the contour plot for $f(x_1,x_2)$.

1. $f(x_1,x_2)$ **must be minimized, and so we would like the solution to be as close to the origin as possible;**

2. The solution must be on the thick line in the right-hand corner of the plot, since this line represents the equality constraint.

Solving a convex program: graphical analysis

.



Solution:

$$\underline{x}^* = (x_1, x_2)^* \approx (1.25, 1.25)$$
$$f(x_1, x_2)^* = 3$$

Any contour *f<3* does not intersect the equality constraint;
Any contour *f>3* intersects the equality constraint at two points.

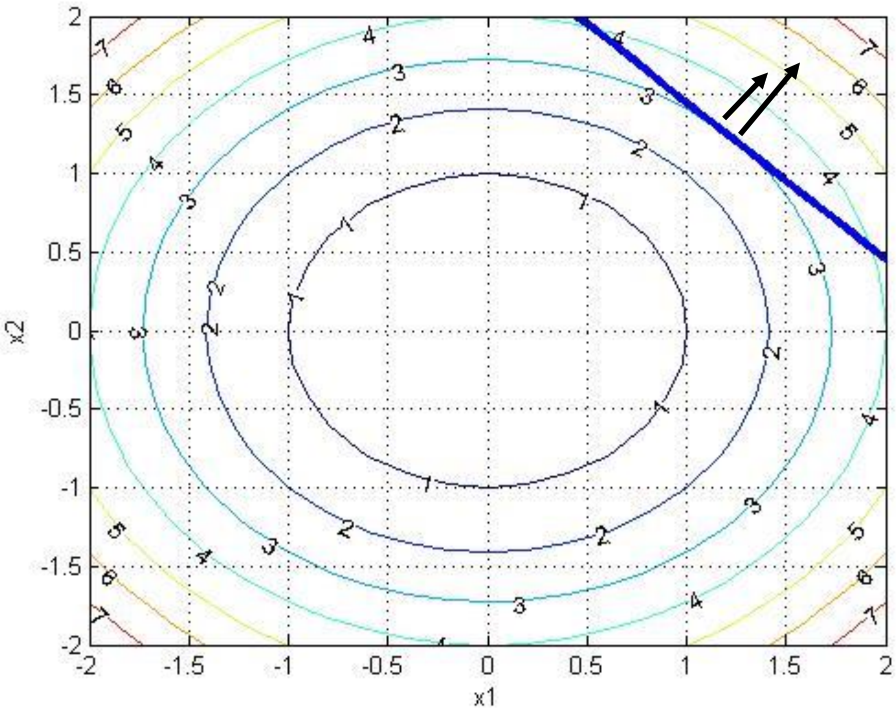
➔The contour f=3 and the equality constraint just touch each other at the point x̲*.

"Just touch":
The two curves are <u>tangent</u> to one another at the solution point.

Solving a convex program: graphical analysis

The two curves are <u>tangent</u> to one another at the solution point.

.



➔ *The normal (gradient) vectors of the two curves, at the solution (tangent) point, are parallel.*

This means the following two vectors are parallel:

$$\nabla\{f(x_1,x_2)^*\} = \nabla\left\{x_1^2 + x_2^2\right\}^* = \begin{bmatrix} 2x_1 \\ 2x_2 \end{bmatrix}^*$$

$$\nabla\{h(x_1,x_2)^* - \sqrt{6}\} = \nabla\left\{x_1 + x_2 - \sqrt{6}\right\}^* = \begin{bmatrix} 1 \\ 1 \end{bmatrix}^*$$

"Parallel" means that the two vectors have the same direction. We do not know that they have the same magnitude. To account for this, we equate with a "multiplier" λ:

$$\nabla f(x_1,x_2)^* = \lambda\nabla\left(h(x_1,x_2)^* - c\right)$$

Solving a convex program: graphical analysis

$$\nabla f(x_1, x_2)^* = \lambda \nabla \left(h(x_1, x_2)^* - c \right)$$

.

Moving everything to the left:

$$\nabla f(x_1, x_2)^* - \lambda \nabla \left(h(x_1, x_2)^* - c \right) = 0$$

Alternately:

$$\nabla f(x_1, x_2)^* + \lambda \nabla \left(c - h(x_1, x_2)^* \right) = 0$$

Performing the gradient operation (taking derivatives with respect to x_1 and x_2) :

$$\begin{bmatrix} \dfrac{\partial}{\partial x_1} \left(f(x_1, x_2) - \lambda \left(h(x_1, x_2) - c \right) \right) \\ \dfrac{\partial}{\partial x_2} \left(f(x_1, x_2) - \lambda \left(h(x_1, x_2) - c \right) \right) \end{bmatrix}^* = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

In this problem, we already know the solution, but what if we did not? Then could we use the above equations to find the solution?

Solving a convex program: analytical analysis

In this problem, we already know the solution, but what if we did not? Then could we use the above equations to find the solution?

$$\begin{bmatrix} \frac{\partial}{\partial x_1}\left(f(x_1,x_2)-\lambda\left(h(x_1,x_2)-c\right)\right) \\ \frac{\partial}{\partial x_2}\left(f(x_1,x_2)-\lambda\left(h(x_1,x_2)-c\right)\right) \end{bmatrix}^* = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

NO! Because we only have 2 equations, yet 3 unknowns: x_1, x_2, λ.
So we need another equation. Where do we get that equation?

Recall our equality constraint: $h(x_1, x_2)-c=0$. **This must be satisfied!**
Therefore:

$$\begin{bmatrix} \frac{\partial}{\partial x_1}\left(f(x_1,x_2)-\lambda\left(h(x_1,x_2)-c\right)\right) \\ \frac{\partial}{\partial x_2}\left(f(x_1,x_2)-\lambda\left(h(x_1,x_2)-c\right)\right) \\ h(x_1,x_2)-c \end{bmatrix}^* = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

Three equations, three unknowns, we can solve.

Solving a convex program: analytical analysis

Observation: The three equations are simply partial derivatives of the function

$$f(x_1, x_2) - \lambda\big(h(x_1,x_2) - c\big)$$

$$\begin{bmatrix} \dfrac{\partial}{\partial x_1}\big(f(x_1, x_2) - \lambda\big(h(x_1,x_2) - c\big)\big) \\ \dfrac{\partial}{\partial x_2}\big(f(x_1, x_2) - \lambda\big|h(x_1,x_2) - c\big|\big) \\ h(x_1, x_2) - c \\ \end{bmatrix}^* = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

This is obviously true for the first two equations , but it is not so obviously true for the last one. But to see it, observe

$$\frac{\partial}{\partial \lambda}\big(f(x_1, x_2) - \lambda\big(h(x_1,x_2) - c\big)\big) = 0$$

$$\Rightarrow -h(x_1,x_2) + c = 0 \Rightarrow h(x_1,x_2) = c$$

Formal approach to solving our problem

Define the Lagrangian function:

$$\mathcal{L}(x_1, x_2, \lambda) = f(x_1, x_2) - \lambda\big(h(x_1,x_2) - c\big)$$

In a convex programming problem, the "first-order conditions" for finding the solution is given by

$$\nabla \mathcal{L}(x_1, x_2, \lambda) = 0$$

OR

$$\frac{\partial}{\partial x_1}\mathcal{L}(x_1, x_2, \lambda) = 0$$

$$\frac{\partial}{\partial x_2}\mathcal{L}(x_1, x_2, \lambda) = 0$$

$$\frac{\partial}{\partial \lambda}\mathcal{L}(x_1, x_2, \lambda) = 0$$

Or more compactly

$$\frac{\partial}{\partial \underline{x}}\mathcal{L}(\underline{x}, \lambda) = 0$$

$$\frac{\partial}{\partial \lambda}\mathcal{L}(\underline{x}, \lambda) = 0$$

where we have used $\underline{\mathbf{x}}$=($\mathbf{x_1}$, $\mathbf{x_2}$)

Applying to our example

Define the Lagrangian function:

$$\mathcal{L}(x_1, x_2, \lambda) = f(x_1, x_2) - \lambda\big(h(x_1, x_2) - c\big)$$
$$= x_1^2 + x_2^2 - \lambda\left(x_1 + x_2 - \sqrt{6}\right)$$

$$\nabla \mathcal{L}(x_1, x_2, \lambda) = 0$$

OR

$$\frac{\partial}{\partial x_1}\mathcal{L}(x_1, x_2, \lambda) = 0 \qquad \frac{\partial}{\partial x_1}\mathcal{L}(x_1, x_2, \lambda) = 2x_1 - \lambda = 0$$

$$\frac{\partial}{\partial x_2}\mathcal{L}(x_1, x_2, \lambda) = 0 \quad \Rightarrow \quad \frac{\partial}{\partial x_2}\mathcal{L}(x_1, x_2, \lambda) = 2x_2 - \lambda = 0$$

$$\frac{\partial}{\partial \lambda}\mathcal{L}(x_1, x_2, \lambda) = 0 \qquad \frac{\partial}{\partial \lambda}\mathcal{L}(x_1, x_2, \lambda) = -\left(x_1 + x_2 - \sqrt{6}\right) = 0$$

A set of 3 linear equations and 3 unknowns; we can write in the form of Ax=b.

Applying to our example

$$\begin{bmatrix} 2 & 0 & -1 \\ 0 & 2 & -1 \\ 1 & 1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \lambda \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \sqrt{6} \end{bmatrix}$$

$$\Rightarrow \begin{bmatrix} x_1 \\ x_2 \\ \lambda \end{bmatrix} = \begin{bmatrix} 2 & 0 & -1 \\ 0 & 2 & -1 \\ 1 & 1 & 0 \end{bmatrix}^{-1} \begin{bmatrix} 0 \\ 0 \\ \sqrt{6} \end{bmatrix} = \begin{bmatrix} 1.2247 \\ 1.2247 \\ 2.4495 \end{bmatrix}$$

Now, let’s go back to our example with a nonlinear equality constraint.

Example with nonlinear equality

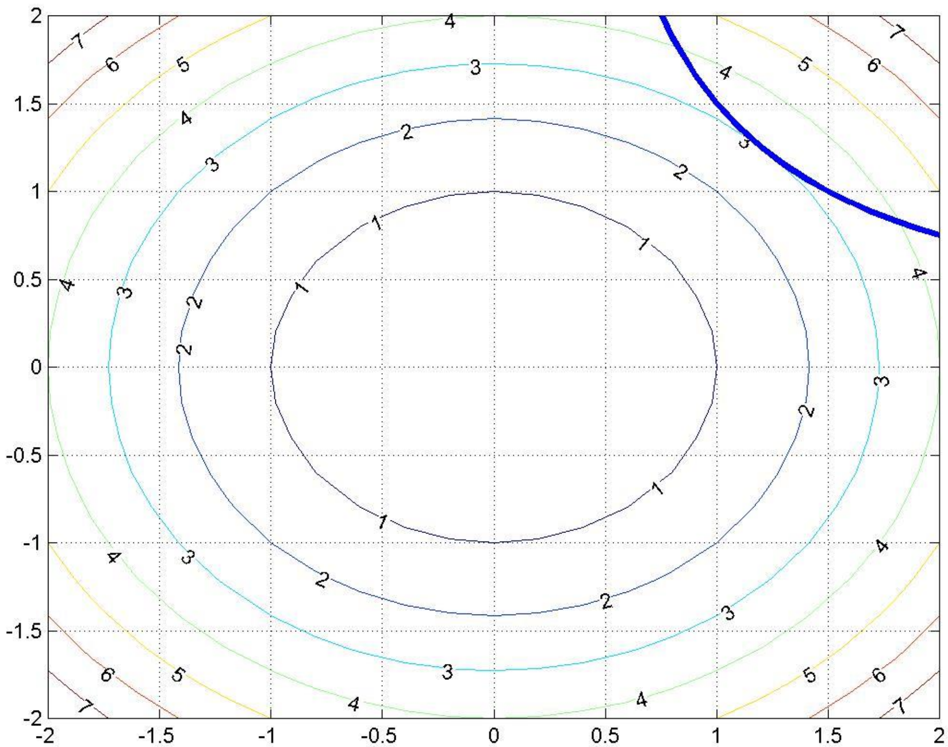
Non-convex because a line connecting two points in the set do not remain in the set. (see "notes" of this slide)

$$\min \ f(x_1, x_2) = x_1^2 + x_2^2$$

$$s.t. \quad h(x_1, x_2) = 2x_1x_2 = 3$$

$$2x_1x_2 = 3 \Rightarrow x_2 = \frac{3}{2x_1}$$

Superimpose this relation on top of the contour plot for f(x_1,x_2).

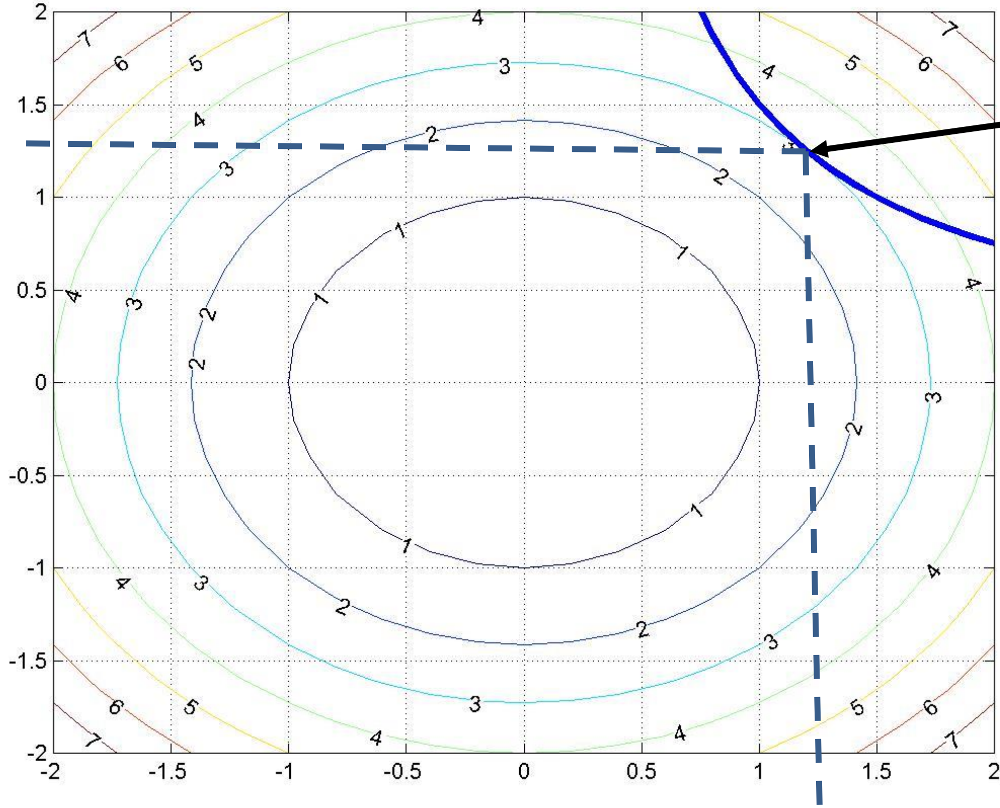


***1. f(x_1,x_2)* must be minimized, and so we would like the solution to be as close to the origin as possible;**

2. The solution must be on the thick line in the right-hand corner of the plot, since this line represents the equality constraint.

Example with nonlinear equality

.



Solution:

$$\underline{x}^* = (x_1, x_2)^* \approx (1.25, 1.25)$$
$$f(x_1, x_2)^* = 3$$

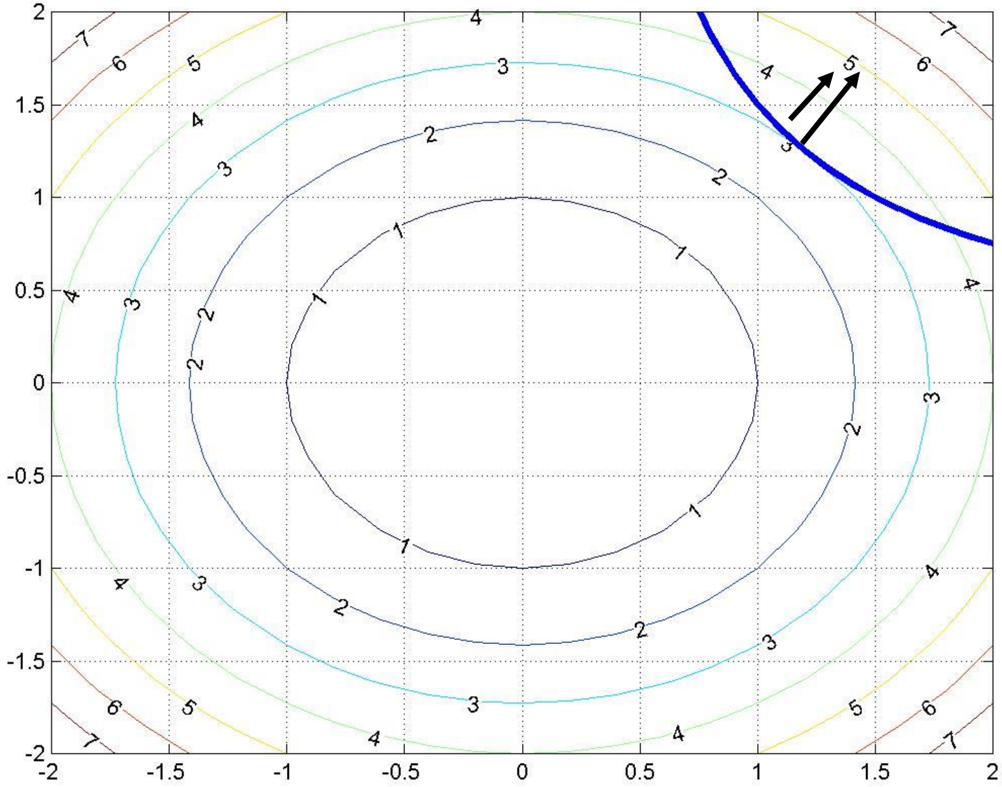
Any contour *f<3* does not intersect the equality constraint;
Any contour *f>3* intersects the equality constraint at two points.

➔The contour f=3 and the equality constraint just touch each other at the point $\underline{\mathbf{x}}$*.

"Just touch":
The two curves are <u>tangent</u> to one another at the solution point.

Example with nonlinear equality

The two curves are <u>tangent</u> to one another at the solution point.



➔ *The normal (gradient) vectors of the two curves, at the solution (tangent) point, are parallel.*

This means the following two vectors are parallel:

$$\nabla\{f(x_1,x_2)^*\} = \nabla\left\{x_1^2 + x_2^2\right\}^* = \begin{bmatrix} 2x_1 \\ 2x_2 \end{bmatrix}^*$$

$$\nabla\{h(x_1,x_2)^* - 3\} = \nabla\left\{2x_1x_2 - 3\right\}^* = \begin{bmatrix} 2x_2 \\ 2x_1 \end{bmatrix}^*$$

"Parallel" means that the two vectors have the same direction. We do not know that they have the same magnitude. To account for this, we equate with a "multiplier" λ:

$$\nabla f(x_1,x_2)^* = \lambda\nabla\left(h(x_1,x_2)^* - c\right)$$

Example with nonlinear equality

This gives us the following two equations.

$$\begin{bmatrix} \dfrac{\partial}{\partial x_1}\left(f(x_1, x_2) - \lambda\left(h(x_1,x_2) - c\right)\right) \\ \dfrac{\partial}{\partial x_2}\left(f(x_1, x_2) - \lambda\left(h(x_1,x_2) - c\right)\right) \end{bmatrix}^* = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

And we add the equality constraint to give 3 equations, 3 unknowns:

$$\begin{bmatrix} \dfrac{\partial}{\partial x_1}\left(f(x_1, x_2) - \lambda\left(h(x_1,x_2) - c\right)\right) \\ \dfrac{\partial}{\partial x_2}\left(f(x_1, x_2) - \lambda\left(h(x_1,x_2) - c\right)\right) \\ h(x_1, x_2) - c \end{bmatrix}^* = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

Three equations, three unknowns, we can solve.

Example with nonlinear equality

Define the Lagrangian function:

$$\mathcal{L}(x_1, x_2, \lambda) = f(x_1, x_2) - \lambda\left(h(x_1, x_2) - c\right)$$

$$= x_1^2 + x_2^2 - \lambda\left(2x_1x_2 - 3\right)$$

$$\nabla\mathcal{L}(x_1, x_2, \lambda) = 0$$

OR

$$\frac{\partial}{\partial x_1}\mathcal{L}(x_1, x_2, \lambda) = 0 \qquad \frac{\partial}{\partial x_1}\mathcal{L}(x_1, x_2, \lambda) = 2x_1 - 2\lambda x_2 = 0$$

$$\frac{\partial}{\partial x_2}\mathcal{L}(x_1, x_2, \lambda) = 0 \quad \Rightarrow \quad \frac{\partial}{\partial x_2}\mathcal{L}(x_1, x_2, \lambda) = 2x_2 - 2\lambda x_1 = 0$$

$$\frac{\partial}{\partial \lambda}\mathcal{L}(x_1, x_2, \lambda) = 0 \qquad \frac{\partial}{\partial \lambda}\mathcal{L}(x_1, x_2, \lambda) = -\left(2x_1x_2 - 3\right) = 0$$

You can solve this algebraically to obtain

$$x_1 = x_2 = \sqrt{\frac{3}{2}} = 1.2247$$

$$x_1 = x_2 = -\sqrt{\frac{3}{2}} = -1.2247$$

and f=3 in both cases

Example with nonlinear equality

Our approach worked in this case, i.e., we found a local optimal point that was also a global optimal point, but because it was not a convex programming problem, we had no guarantee that this would happen.

The conditions we established, below, we call first order conditions.

For convex programming problems, they are first order *sufficient conditions* to provide the global optimal point.

For nonconvex programming problems, they are first order *necessary conditions* to provide the global optimal point.

$$\frac{\partial}{\partial \underline{x}} \mathcal{L}(\underline{x}, \lambda) = 0$$

$$\frac{\partial}{\partial \lambda} \mathcal{L}(\underline{x}, \lambda) = 0$$

Multiple equality constraints

$$\min\ f(\underline{x})$$
$$s.t.\quad \underline{h}(\underline{x}) = \underline{c}$$

We assume that f and h are continuously differentiable.

$$\mathcal{L}(\underline{x},\underline{\lambda}) = f(\underline{x}) - \lambda_1\big(h_1(\underline{x}) - c_1\big) - \lambda_2\big(h_2(\underline{x}) - c_2\big) - \ldots - \lambda_m\big(h_m(\underline{x}) - c_m\big)$$

First order necessary conditions that ($\underline{\mathbf{x}}$*, $\underline{\boldsymbol{\lambda}}$*) solves the above:

$$\frac{\partial}{\partial \underline{x}}\mathcal{L}(\underline{x}^*,\underline{\lambda}^*) = 0$$

$$\frac{\partial}{\partial \underline{\lambda}}\mathcal{L}(\underline{x}^*,\underline{\lambda}^*) = 0$$

Multiple equality & 1 inequality constraint

$$\begin{aligned} \min \ & f(\underline{x}) \\ s.t. \ & \underline{h}(\underline{x}) = \underline{c} \\ & g(\underline{x}) \geq b \end{aligned}$$

We assume that f, h, and g are continuously differentiable.

Solution approach:

- **Ignore the inequality constraint and solve the problem. (this is just a problem with multiple equality constraints).**
- **If inequality constraint is satisfied, then problem is solved.**
- **If inequality constraint is violated, then the inequality constraint must be <u>binding</u> ➔ inequality constraint enforced with equality:**

$$g(\underline{x}) = b$$

Let's look at this new problem where the inequality is binding.

Multiple equality & 1 inequality constraint

$$\min \ f(\underline{x})$$
$$s.t. \ \ \underline{h}(\underline{x}) = \underline{c}$$
$$g(\underline{x}) = b$$

We assume that f, h, and g are continuously differentiable.

$$\mathcal{L}(\underline{x}, \underline{\lambda}, \mu) = f(\underline{x}) - \lambda_1\left(h_1(\underline{x}) - c_1\right) - \lambda_2\left(h_2(\underline{x}) - c_2\right) - \ldots - \lambda_m\left(h_m(\underline{x}) - c_m\right) - \mu\left(g(\underline{x}) - b\right)$$

First order necessary conditions that ($\underline{\mathbf{x}}$*, $\underline{\boldsymbol{\lambda}}$*, μ*) solves the above:

$$\frac{\partial}{\partial \underline{x}} \mathcal{L}(\underline{x}^*, \underline{\lambda}^*, \mu^*) = 0$$

$$\frac{\partial}{\partial \underline{\lambda}} \mathcal{L}(\underline{x}^*, \underline{\lambda}^*, \mu^*) = 0$$

$$\frac{\partial}{\partial \mu} \mathcal{L}(\underline{x}^*, \underline{\lambda}^*, \mu^*) = 0$$

We were able to write down this solution only after we knew the inequality constraint was binding. Can we generalize this approach?

Multiple equality & 1 inequality constraint

$$\mathcal{L}(\underline{x},\underline{\lambda},\mu) = f(\underline{x}) - \lambda_1\left(h_1(\underline{x}) - c_1\right) - \lambda_2\left(h_2(\underline{x}) - c_2\right) - \ldots - \lambda_m\left(h_m(\underline{x}) - c_m\right) - \mu\left(g(\underline{x}) - b\right)$$

If inequality is not binding, then apply first order necessary conditions by ignoring it:

➔μ=0

➔g(x)-b≠0 (since it is not binding!)

If inequality is binding, then apply first order necessary conditions treating inequality constraint as an equality constraint

➔μ≠0

➔g(x)-b≠0 (since it is binding!)

Either way:
μ(g(x)-b)=0

This relation encodes our solution procedure!
It can be used to generalize our necessary conditions

Multiple equality & multiple inequality constraints

$$\min \; f(\underline{x})$$
$$s.t. \;\; \underline{h}(\underline{x}) = \underline{c}$$
$$\underline{g}(\underline{x}) = \underline{b}$$

We assume that f, h, and g are continuously differentiable.

$$\mathcal{L}(\underline{x},\underline{\lambda},\underline{\mu}) = f(\underline{x}) - \lambda_1\big(h_1(\underline{x}) - c_1\big) - \lambda_2\big(h_2(\underline{x}) - c_2\big) \text{-} \ldots - \lambda_m\big(h_m(\underline{x}) - c_m\big)$$
$$- \mu_1\big(g_1(\underline{x}) - b_1\big) - \mu_2\big(g_1(\underline{x}) - b_1\big) - \ldots - \mu_n\big(g_1(\underline{x}) - b_1\big)$$

First order necessary conditions that (x*, λ*, μ*) solves the above:

$$\frac{\partial}{\partial \underline{x}}\mathcal{L}(\underline{x}^*,\underline{\lambda}^*,\underline{\mu}^*) = 0$$
$$\frac{\partial}{\partial \underline{\lambda}}\mathcal{L}(\underline{x}^*,\underline{\lambda}^*,\underline{\mu}^*) = 0$$
$$\frac{\partial}{\partial \underline{\mu}}\mathcal{L}(\underline{x}^*,\underline{\lambda}^*,\underline{\mu}^*) = 0$$
$$\mu_k^*(g_k(\underline{x}^*) - b) = 0 \qquad \forall k$$
$$\mu_k^* \geq 0 \qquad \forall k$$

These conditions also referred to as the Kurash-Kuhn-Tucker (KKT) conditions

Complementarity condition: Inactive constraints have a zero multiplier.

Nonnegativity on inequality multipliers.

An additional requirement

$$\min \ f(\underline{x})$$
$$s.t. \quad \underline{h}(\underline{x}) = \underline{c}$$
$$\underline{g}(\underline{x}) = \underline{b}$$

We assume that f, h, and g are continuously differentiable.

For KKT to guarantee finds a local optimum, we need the Kuhn-Tucker Constraint Qualification (even under convexity).

This condition imposes a certain restriction on the constraint functions .

Its purpose is to rule out certain irregularities on the boundary of the feasible set, that would invalidate the Kuhn-Tucker conditions should the optimal solution occur there.

We will not try to tackle this idea, but know this:
➔If the feasible region is a convex set formed by *linear* constraints only, then the constraint qualification will be met, and the Kuhn-Tucker conditions will always hold at an optimal solution.

Economic dispatch calculation (EDC)

Generator unit cost function:

$\mathrm{COST}_i =$ $C_i(P_i) = a_i(P_i)^2 + b_i P_i + c_i$

where

COST_i = production cost

P_i = production power

Power balance

$$\sum_{i=1}^{n} P_i = P_D + P_{LOSS} + P_{tie} = P_T$$

(no transmission representation)

Unit capacity limits

$$P_i \geq \underline{\underline{P}}_i \qquad P_i \leq \overline{\overline{P}}_i$$

$where:$

$$\underline{\underline{P}}_i = P_{\min} = \min \textit{generation level}$$

$$\overline{\overline{P}}_i = P_{\max} = \max \textit{generation level}$$

Notation: double underline means lower bound.

Double overline means upper bound.

General EDC problem statement.

$$\min f(P_i) = \sum_{i=1}^{n} C_i(P_i)$$

Subject to

$$\sum_{i=1}^{n} P_i = P_T$$

$$P_i \geq \underline{\underline{P}}_i$$

$$P_i \leq \overline{\overline{P}}_i \Rightarrow -P_i \geq -\overline{\overline{P}}_i$$

Two unit system, Lagrangian function:

$$\begin{aligned}
\mathcal{L}\left(P_1, P_2, \lambda, \underline{\underline{\mu}}_1, \overline{\overline{\mu}}_1, \underline{\underline{\mu}}_2, \overline{\overline{\mu}}_2\right) = \; & C_1(P_1) + C_2(P_2) \\
& -\lambda\left[P_1 + P_2 - P_T\right] \\
& -\underline{\underline{\mu}}_1\left[P_1 - \underline{\underline{P}}_1\right] - \overline{\overline{\mu}}_1\left[-P_1 + \overline{\overline{P}}_1\right] \\
& -\underline{\underline{\mu}}_2\left[P_2 - \underline{\underline{P}}_2\right] - \overline{\overline{\mu}}_2\left[-P_2 + \overline{\overline{P}}_2\right]
\end{aligned}$$

Two unit system, KKT conditions:

$$\frac{\partial \mathcal{L}}{\partial P_1} = 0 \quad \Rightarrow \quad \frac{\partial C_1(P_1)}{\partial P_1} - \lambda - \underline{\underline{\mu}}_1 + \overline{\overline{\mu}}_1 = 0$$

$$\frac{\partial \mathcal{L}}{\partial P_2} = 0 \quad \Rightarrow \quad \frac{\partial C_2(P_2)}{\partial P_2} - \lambda - \underline{\underline{\mu}}_2 + \overline{\overline{\mu}}_2 = 0$$

$$\frac{\partial \mathcal{L}}{\partial \lambda} = 0 \quad \Rightarrow \quad P_1 + P_2 - P_T = 0$$

$$\underline{\mu}^{\mathrm{T}}\left[\underline{h}(\underline{x}) - \underline{c}\right] = \underline{0} \quad \Rightarrow \quad \underline{\underline{\mu}}_1\left[P_1 - \underline{\underline{P}}_1\right] = 0, \;\; \overline{\overline{\mu}}_1\left[-P_1 + \overline{\overline{P}}_1\right] = 0$$

$$\underline{\underline{\mu}}_2\left[P_2 - \underline{\underline{P}}_2\right] = 0, \;\; \overline{\overline{\mu}}_2\left[-P_2 + \overline{\overline{P}}_2\right] = 0$$

$$\underline{\underline{\mu}}_k \geq 0, k = 1,2$$

$$\overline{\overline{\mu}}_k \geq 0, k = 1,2$$

	Unit 1	Unit 2
Generation Specifications:		
Minimum Generation	200 MW	100 MW
Maximum Generation	380 MW	200 MW
Cost Curve Coefficients:		
Quadratic Term	0.016	0.019
Linear Term	2.187	2.407
Constant Term	120.312	74.074

$$C_1(P_1) = 0.016(P_1)^2 + 2.187(P_1) + 120.312$$

$$C_2(P_2) = 0.019(P_2)^2 + 2.407(P_2) + 74.074$$

$$h(P_1, P_2) = P_1 + P_2 = 400$$

$$200 \le P_1 \le 380 \Rightarrow -P_1 \ge -380,\ \ P_1 \ge 200$$

$$100 \le P_2 \le 200 \Rightarrow -P_2 \ge -200,\ \ P_2 \ge 100$$

LaGrangian function

$$\mathcal{L}\left(P_1, P_2, \lambda, \underline{\underline{\mu}}_1, \overline{\overline{\mu}}_1, \underline{\underline{\mu}}_2, \overline{\overline{\mu}}_2\right) = 0.016\ (P_1)^2 + 2.187\ (P_1) + 120.312$$

$$+ 0.019\,(P_2)^2 + 2.407\,(P_2) + 74.074$$

$$- \lambda\left[P_1 + P_2 - 400\right]$$

$$- \underline{\underline{\mu}}_1\left[P_1 - 200\right] - \overline{\overline{\mu}}_1\left[- P_1 + 380\right]$$

$$- \underline{\underline{\mu}}_2\left[P_2 - 100\right] - \overline{\overline{\mu}}_2\left[- P_2 + 200\right]$$

KKT conditions

$$\frac{\partial \mathcal{L}}{\partial P_1} = 0 \quad \Rightarrow \quad 0.032(P_1) + 2.187 - \lambda + \underline{\underline{\mu}}_1 - \overline{\overline{\mu}}_1 = 0$$

$$\frac{\partial \mathcal{L}}{\partial P_2} = 0 \quad \Rightarrow \quad 0.038(P_2) + 2.407 - \lambda + \underline{\underline{\mu}}_2 - \overline{\overline{\mu}}_2 = 0$$

$$\frac{\partial \mathcal{L}}{\partial \lambda} = 0 \quad \Rightarrow \quad P_1 + P_2 - 400 = 0$$

$$\underline{\mu}^{\mathrm{T}}\left[\underline{g}(\underline{x}) - \underline{c}\right] = \underline{0} \quad \Rightarrow \quad \underline{\underline{\mu}}_1\left[P_1 - 200\right] = 0,\ \ \overline{\overline{\mu}}_1\left[- P_1 + 380\right] = 0$$

$$\underline{\underline{\mu}}_2\left[P_2 - 100\right] = 0,\ \ \overline{\overline{\mu}}_2\left[- P_2 + 200\right] = 0$$

Assume all inequality constraints are non-binding.

This means that

$$\underline{\underline{\mu}}_i = 0 \qquad and \qquad \overline{\overline{\mu}}_i = 0 \qquad\qquad \forall\, i = 1, n$$

And KKT conditions become

$$0.032(P_1) + 2.187 - \lambda = 0$$

$$0.038(P_2) + 2.407 - \lambda = 0$$

$$P_1 + P_2 - 400 = 0$$

Rewrite them as:

$$0.032(P_1) + 0(P_2) - \lambda = -2.187$$

$$(P_1) + 0.038(P_2) - \lambda = -2.407$$

$$P_1 + P_2 = 400$$

And it is easy to see how to put them into matrix form for solution in matlab.

$$\begin{bmatrix} 0.032 & 0 & -1 \\ 0 & 0.038 & -1 \\ 1 & 1 & 0 \end{bmatrix} \times \begin{bmatrix} P_1 \\ P_2 \\ \lambda \end{bmatrix} = \begin{bmatrix} -2.187 \\ -2.407 \\ 400 \end{bmatrix}$$

Solution yields:

$$\begin{bmatrix} P_1 \\ P_2 \\ \lambda \end{bmatrix} = \begin{bmatrix} 220.29 \\ 179.71 \\ 9.24 \end{bmatrix}$$

What is λ = $9.24/MW-hr ???

It is the <u>system</u> “incremental cost.”

It is the cost if the system provides an additional MW over the next hour.

It is the cost of “increasing” the RHS of the equality constraint by 1 MW for an hour.

We can verify this.

Verification for meaning of lambda.

- Compute total costs/hr for Pd=400 MW
- Compute total costs/hr for Pd=401 MW
- Find the difference in total costs/hr for the two demands.

If our interpretation of lambda is correct, this difference should be $9.24.

Get cost/hr for each unit.

$$C_1(P_1) = 0.016(220.29)^2 + 2.187(220.29) + 120.312$$

$$C_1(P_1) = 1378.53\ \$/hr$$

$$C_2(P_2) = 0.019(179.71)^2 + 2.407(179.71) + 74.074$$

$$C_2(P_2) = 1120.25\ \$/hr$$

Total cost/hr are C1+C2

$$C_T = C_1(P_1) + C_2(P_2) = 1378.53 + 1120.25 = 2498.78$$

Now solve EDC for Pd=401 MW to get P1,P2

$$\begin{bmatrix} 0.032 & 0 & -1 \\ 0 & 0.038 & -1 \\ 1 & 1 & 0 \end{bmatrix} \times \begin{bmatrix} P_1 \\ P_2 \\ \lambda \end{bmatrix} = \begin{bmatrix} -2.187 \\ -2.407 \\ 401 \end{bmatrix}$$

$$\begin{bmatrix} P_1 \\ P_2 \\ \lambda \end{bmatrix} = \begin{bmatrix} 220.83 \\ 180.17 \\ 9.25 \end{bmatrix}$$

Get cost/hr for each unit.

$$C_1(P_1) = 0.016(220.83)^2 + 2.187(220.83) + 120.312$$

$$C_1(P_1) = 1383.52\ \$/hr$$

$$C_2(P_2) = 0.019(180.17)^2 + 2.407(180.17) + 74.074$$

$$C_2(P_2) = 1124.51\ \$/hr$$

Total cost/hr are C1+C2

$$C_T = C_1(P_1) + C_2(P_2) = 1383.52 + 1124.51 = 2508.03$$

Total cost/hr changed by 2508.03-2498.78 = 9.25 \$/hr, which is in agreement with our interpretation of lambda.