Optimization methods
Lecture 5: Convex modeling

Alexandr Katrutsa

Modern State of Artificial Intelligence Masters Program
Moscow Institute of Physics and Technology

Brief reminder of the previous lecture

- Dual function and dual problem

Brief reminder of the previous lecture

- Dual function and dual problem
- Weak and strong duality

Brief reminder of the previous lecture

- Dual function and dual problem
- Weak and strong duality
- Slater regularity condition

Plan for today

- General approaches to deal with convexity

Plan for today

- General approaches to deal with convexity
- Standard forms of convex optimization problems

Plan for today

- General approaches to deal with convexity
- Standard forms of convex optimization problems
- Convex calculus

Plan for today

- General approaches to deal with convexity
- Standard forms of convex optimization problems
- Convex calculus
- Disciplined convex programming

Optimization problem

$$
\begin{aligned}
&\min_{\mathbf{x} \in \mathbb{R}^n} f_0(\mathbf{x}) \\
\text{s.t. } & f_i(\mathbf{x}) \leq 0,\ i = 1, \ldots, m \\
& h_j(\mathbf{x}) = 0,\ j = 1, \ldots, p
\end{aligned}
$$

- The possibility of efficient solving depends on the properties f_0, f_i, h_j

Optimization problem

$$\begin{aligned} &\min_{\mathbf{x}\in\mathbb{R}^n} f_0(\mathbf{x}) \\ \text{s.t. } & f_i(\mathbf{x}) \leq 0,\ i = 1,\dots,m \\ & h_j(\mathbf{x}) = 0,\ j = 1,\dots,p \end{aligned}$$

- The possibility of efficient solving depends on the properties f_0, f_i, h_j
- If f_0, f_i, h_j are affine, then we have a linear programming problem (LP), that can be solved extremely fast

Optimization problem

$$\min_{\mathbf{x}\in\mathbb{R}^n} f_0(\mathbf{x})$$
$$\text{s.t. } f_i(\mathbf{x}) \leq 0,\ i = 1,\ldots,m$$
$$h_j(\mathbf{x}) = 0,\ j = 1,\ldots,p$$

- The possibility of efficient solving depends on the properties f_0, f_i, h_j
- If f_0, f_i, h_j are affine, then we have a linear programming problem (LP), that can be solved extremely fast
- Simple problems with non-linear inequality constraints f_i, h_j may be very difficult to solve

Convex optimization problem

$$\begin{aligned} &\min_{\mathbf{x}\in\mathbb{R}^n} f_0(\mathbf{x}) \\ \text{s.t. } &f_i(\mathbf{x}) \leq 0,\ i = 1, \ldots, m \\ &\mathbf{A}\mathbf{x} = \mathbf{b} \end{aligned}$$

- f_0, f_i are convex functions: for all $\mathbf{x}, \mathbf{y}$ and $\alpha \in [0, 1]$

$$f(\alpha\mathbf{x} + (1-\alpha)\mathbf{y}) \leq \alpha f(\mathbf{x}) + (1-\alpha) f(\mathbf{y})$$

- Equality constraints are affine

Convex optimization problems features

- They form a subset of optimization problems: LP is a particular case

Convex optimization problems features

- They form a subset of optimization problems: LP is a particular case
- Their form can be very complex, but the complexity of solving is asymptotically the same as in the LP case

Convex optimization problems features

- They form a subset of optimization problems: LP is a particular case
- Their form can be very complex, but the complexity of solving is asymptotically the same as in the LP case
- Many applications

General approaches to convexity utilizing

- Hope/assume/pretend, that f_i are convex

General approaches to convexity utilizing

- Hope/assume/pretend, that f_i are convex
  - Easy to use

General approaches to convexity utilizing

- Hope/assume/pretend, that f_i are convex
  - Easy to use
  - Loose some benefits from convexity

General approaches to convexity utilizing

- Hope/assume/pretend, that f_i are convex
  - Easy to use
  - Loose some benefits from convexity
- Check convexity before solving

General approaches to convexity utilizing

- Hope/assume/pretend, that f_i are convex
  - Easy to use
  - Loose some benefits from convexity
- Check convexity before solving
  - very difficult in general case

General approaches to convexity utilizing

- Hope/assume/pretend, that f_i are convex
  - Easy to use
  - Loose some benefits from convexity
- Check convexity before solving
  - very difficult in general case
- Construction of convex problems from elementary blocks

General approaches to convexity utilizing

- Hope/assume/pretend, that f_i are convex
  - Easy to use
  - Loose some benefits from convexity
- Check convexity before solving
  - very difficult in general case
- Construction of convex problems from elementary blocks
  - user follows the fixed set of rules to set up f_i

General approaches to convexity utilizing

- Hope/assume/pretend, that f_i are convex
  - Easy to use
  - Loose some benefits from convexity
- Check convexity before solving
  - very difficult in general case
- Construction of convex problems from elementary blocks
  - user follows the fixed set of rules to set up f_i
  - convexity of the optimization problem is checked automatically

How to verify convexity?

- Definition, criteria, like $f''(\mathbf{x}) \succeq 0$

How to verify convexity?

- Definition, criteria, like $f''(\mathbf{x}) \succeq 0$
- Convex functions calculus: construct f in a special manner

How to verify convexity?

- Definition, criteria, like $f''(\mathbf{x}) \succeq 0$
- Convex functions calculus: construct f in a special manner
  - Given a set of simple convex functions which is known to be convex

How to verify convexity?

- Definition, criteria, like $f''(\mathbf{x}) \succeq 0$
- Convex functions calculus: construct f in a special manner
  - Given a set of simple convex functions which is known to be convex
  - Given a set of composition rules and a proper transformations that are preserved the convexity of result

Examples of simple convex functions

- If $x > 0$: x^p for $p < 0,\ p \geq 1$ and x^{-p} if $p \in [0, 1]$

Examples of simple convex functions

- If $x > 0$: x^p for $p < 0,\ p \geq 1$ and x^{-p} if $p \in [0, 1]$
- $e^x,\ -\log x,\ x \log x$

Examples of simple convex functions

- If $x > 0$: x^p for $p < 0,\ p \geq 1$ and x^{-p} if $p \in [0, 1]$
- $e^x,\ -\log x,\ x \log x$
- $\langle \mathbf{a}, \mathbf{x} \rangle + b$

Examples of simple convex functions

- If $x > 0$: x^p for $p < 0,\ p \geq 1$ and x^{-p} if $p \in [0, 1]$
- $e^x,\ -\log x,\ x \log x$
- $\langle \mathbf{a}, \mathbf{x} \rangle + b$
- $\|\mathbf{x}\|$ – any norm

Examples of simple convex functions

- If $x > 0$: x^p for $p < 0,\ p \geq 1$ and x^{-p} if $p \in [0, 1]$
- $e^x,\ -\log x,\ x \log x$
- $\langle \mathbf{a}, \mathbf{x} \rangle + b$
- $\|\mathbf{x}\|$ – any norm
- $\max\{x_1, \ldots, x_n\}$ and $\log(e^{x_1} + \ldots + e^{x_n})$

Examples of simple convex functions

- If $x > 0$: x^p for $p < 0,\ p \geq 1$ and x^{-p} if $p \in [0, 1]$
- $e^x,\ -\log x,\ x \log x$
- $\langle \mathbf{a}, \mathbf{x} \rangle + b$
- $\|\mathbf{x}\|$ – any norm
- $\max\{x_1, \ldots, x_n\}$ and $\log(e^{x_1} + \ldots + e^{x_n})$
- $\log \det \mathbf{X}^{-1}$ for $\mathbf{X} \in \mathbb{S}^n_+$

Some rules from convex functions calculus

- Multiplication by non-negative constant: f is convex and $\alpha > 0$, then αf is convex

Some rules from convex functions calculus

- Multiplication by non-negative constant: f is convex and $\alpha > 0$, then αf is convex
- Summation: f, g are convex, then $f + g$ is convex

Some rules from convex functions calculus

- Multiplication by non-negative constant: f is convex and $\alpha > 0$, then αf is convex
- Summation: f, g are convex, then $f + g$ is convex
- Composition with affine function: f is convex, then $f(\mathbf{A}\mathbf{x} + \mathbf{b})$ is also convex

Some rules from convex functions calculus

- Multiplication by non-negative constant: f is convex and $\alpha > 0$, then αf is convex
- Summation: f, g are convex, then $f + g$ is convex
- Composition with affine function: f is convex, then $f(\mathbf{A}\mathbf{x} + \mathbf{b})$ is also convex
- Taking maximum: $f_1, \ldots, f_m$ are convex, then $\max\limits_{i=1,\ldots,m} \{f_i(\mathbf{x})\}$ is convex

Some rules from convex functions calculus

- Multiplication by non-negative constant: f is convex and $\alpha > 0$, then αf is convex
- Summation: f, g are convex, then $f + g$ is convex
- Composition with affine function: f is convex, then $f(\mathbf{Ax} + \mathbf{b})$ is also convex
- Taking maximum: $f_1, \ldots, f_m$ are convex, then $\max\limits_{i=1,\ldots,m} \{f_i(\mathbf{x})\}$ is convex
- Composition: if h is convex and increasing, f is convex, then $g(\mathbf{x}) = h(f(\mathbf{x}))$ is convex

Some rules from convex functions calculus

- Multiplication by non-negative constant: f is convex and $\alpha > 0$, then αf is convex
- Summation: f, g are convex, then $f + g$ is convex
- Composition with affine function: f is convex, then $f(\mathbf{A}\mathbf{x} + \mathbf{b})$ is also convex
- Taking maximum: $f_1, \ldots, f_m$ are convex, then $\max\limits_{i=1,\ldots,m} \{f_i(\mathbf{x})\}$ is convex
- Composition: if h is convex and increasing, f is convex, then $g(\mathbf{x}) = h(f(\mathbf{x}))$ is convex
- And many others...

Examples

- $f(\mathbf{x}) = \max_{i=1,\ldots,m} (\langle \mathbf{a}_i, \mathbf{x} \rangle + b_i)$

Examples

- $f(\mathbf{x}) = \max_{i=1,\ldots,m} (\langle \mathbf{a}_i, \mathbf{x} \rangle + b_i)$
- ℓ_1-regularized least-squares problem

$$f(\mathbf{x}) = \frac{1}{2}\|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2^2 + \lambda\|\mathbf{x}\|_1, \quad \lambda > 0$$

Examples

- $f(\mathbf{x}) = \max_{i=1,\ldots,m} (\langle \mathbf{a}_i, \mathbf{x} \rangle + b_i)$
- ℓ_1-regularized least-squares problem

$$f(\mathbf{x}) = \frac{1}{2}\|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2^2 + \lambda\|\mathbf{x}\|_1, \quad \lambda > 0$$

- Logarithmic barrier

$$-\sum_{i=1}^{m} \log(-f_i(\mathbf{x}))$$

for $\{\mathbf{x} \mid f_i(\mathbf{x}) < 0\}$ and convex $f_i(\mathbf{x})$

Examples

- $f(\mathbf{x}) = \max_{i=1,\ldots,m} (\langle \mathbf{a}_i, \mathbf{x} \rangle + b_i)$
- ℓ_1-regularized least-squares problem

$$f(\mathbf{x}) = \frac{1}{2}\|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2^2 + \lambda\|\mathbf{x}\|_1, \quad \lambda > 0$$

- Logarithmic barrier

$$-\sum_{i=1}^{m} \log(-f_i(\mathbf{x}))$$

  for $\{\mathbf{x} \mid f_i(\mathbf{x}) < 0\}$ and convex $f_i(\mathbf{x})$
- Maximum eigenvalue of $\mathbf{A} \in \mathbb{S}^n$:

$$\lambda_{\max}(\mathbf{A}) = \sup_{\|\mathbf{x}\|_2 = 1} (\mathbf{x}^\top \mathbf{A}\mathbf{x})$$

How to solve convex optimization problems?

- Use «standard» solver for specific class of problems

How to solve convex optimization problems?

- Use «standard» solver for specific class of problems
  - easy to run

How to solve convex optimization problems?

- Use «standard» solver for specific class of problems
  - easy to run
  - a problem ***has to be*** in the standard form for the considered solver

How to solve convex optimization problems?

- Use «standard» solver for specific class of problems
  - easy to run
  - a problem ***has to be*** in the standard form for the considered solver
  - many users

How to solve convex optimization problems?

- Use «standard» solver for specific class of problems
  - easy to run
  - a problem ***has to be*** in the standard form for the considered solver
  - many users
- Create and/or implement your own method

How to solve convex optimization problems?

- Use «standard» solver for specific class of problems
  - easy to run
  - a problem ***has to be*** in the standard form for the considered solver
  - many users
- Create and/or implement your own method
  - Time consuming

How to solve convex optimization problems?

- Use «standard» solver for specific class of problems
  - easy to run
  - a problem ***has to be*** in the standard form for the considered solver
  - many users
- Create and/or implement your own method
  - Time consuming
  - May be more efficient for a particular problem

How to solve convex optimization problems?

- Use «standard» solver for specific class of problems
  - easy to run
  - a problem ***has to be*** in the standard form for the considered solver
  - many users
- Create and/or implement your own method
  - Time consuming
  - May be more efficient for a particular problem
- Transform the problem into standard from and use standard solver

How to solve convex optimization problems?

- Use «standard» solver for specific class of problems
  - easy to run
  - a problem ***has to be*** in the standard form for the considered solver
  - many users
- Create and/or implement your own method
  - Time consuming
  - May be more efficient for a particular problem
- Transform the problem into standard from and use standard solver
  - Extend a set of problems that can be treated with standard solvers

How to solve convex optimization problems?

- Use «standard» solver for specific class of problems
  - easy to run
  - a problem ***has to be*** in the standard form for the considered solver
  - many users
- Create and/or implement your own method
  - Time consuming
  - May be more efficient for a particular problem
- Transform the problem into standard from and use standard solver
  - Extend a set of problems that can be treated with standard solvers
  - Transformation may be not simple

Three main classes of problems

- Linear programming

$$\begin{aligned} &\min \mathbf{c}^\top \mathbf{x} \\ \text{s.t. } &\mathbf{A}\mathbf{x} = \mathbf{b} \\ &\mathbf{x} \geq 0 \end{aligned}$$

Three main classes of problems

- Linear programming

$$
\begin{aligned}
& \min \mathbf{c}^\top \mathbf{x} \\
\text{s.t. } & \mathbf{A}\mathbf{x} = \mathbf{b} \\
& \mathbf{x} \geq 0
\end{aligned}
$$

- Second-order cone programming

$$
\begin{aligned}
& \min \mathbf{f}^\top \mathbf{x} \\
\text{s.t. } & \|\mathbf{A}_i \mathbf{x} + \mathbf{b}_i\|_2 \leq \mathbf{c}_i^\top \mathbf{x} + d_i \\
& \mathbf{F}\mathbf{x} = \mathbf{g}
\end{aligned}
$$

Three main classes of problems

- Linear programming

$$\begin{aligned} & \min \mathbf{c}^\top \mathbf{x} \\ \text{s.t. } & \mathbf{A}\mathbf{x} = \mathbf{b} \\ & \mathbf{x} \geq 0 \end{aligned}$$

- Second-order cone programming

$$\begin{aligned} & \min \mathbf{f}^\top \mathbf{x} \\ \text{s.t. } & \|\mathbf{A}_i \mathbf{x} + \mathbf{b}_i\|_2 \leq \mathbf{c}_i^\top \mathbf{x} + d_i \\ & \mathbf{F}\mathbf{x} = \mathbf{g} \end{aligned}$$

- Semidefinite programming problem

$$\begin{aligned} & \min_{\mathbf{X}} \operatorname{trace}(\mathbf{C}\mathbf{X}) \\ \text{s.t. } & \operatorname{trace}(\mathbf{A}_i \mathbf{X}) = b_i \\ & \mathbf{X} \succeq 0 \end{aligned}$$

Interior point method (IPM) for convex problems

- Interior-Point Polynomial Algorithms in Convex Programming, Y. Nesterov, A. Nemirovskii, 1994

Interior point method (IPM) for convex problems

- Interior-Point Polynomial Algorithms in Convex Programming, Y. Nesterov, A. Nemirovskii, 1994
- Review of IPM see here

Interior point method (IPM) for convex problems

- Interior-Point Polynomial Algorithms in Convex Programming, Y. Nesterov, A. Nemirovskii, 1994
- Review of IPM see here
- These family of methods is applicable if f_i is smooth and given problem is in standard form

Interior point method (IPM) for convex problems

- Interior-Point Polynomial Algorithms in Convex Programming, Y. Nesterov, A. Nemirovskii, 1994
- Review of IPM see here
- These family of methods is applicable if f_i is smooth and given problem is in standard form
- Extremely efficient method: some dozens of iterations are sufficient to convergence independently on the problem dimension

Interior point method (IPM) for convex problems

- Interior-Point Polynomial Algorithms in Convex Programming, Y. Nesterov, A. Nemirovskii, 1994
- Review of IPM see here
- These family of methods is applicable if f_i is smooth and given problem is in standard form
- Extremely efficient method: some dozens of iterations are sufficient to convergence independently on the problem dimension
- Every iteration requires solving some linear system

What if IPM is not applicable?

What if IPM is not applicable?

- Example: ℓ_1-regularized least-squares problem

$$f(\mathbf{x}) = \frac{1}{2}\|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2^2 + \lambda\|\mathbf{x}\|_1, \quad \lambda > 0$$

What if IPM is not applicable?

- Example: ℓ_1-regularized least-squares problem

$$f(\mathbf{x}) = \frac{1}{2}\|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2^2 + \lambda\|\mathbf{x}\|_1, \quad \lambda > 0$$

- The problem is convex, but f is non-smooth!

What if IPM is not applicable?

- Example: ℓ_1-regularized least-squares problem

$$f(\mathbf{x}) = \frac{1}{2}\|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2^2 + \lambda\|\mathbf{x}\|_1, \quad \lambda > 0$$

- The problem is convex, but f is non-smooth!
- **Main idea**: transform the problem such that IPM becomes applicable

What if IPM is not applicable?

- Example: ℓ_1-regularized least-squares problem

$$f(\mathbf{x}) = \frac{1}{2}\|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2^2 + \lambda\|\mathbf{x}\|_1, \quad \lambda > 0$$

- The problem is convex, but f is non-smooth!
- **Main idea**: transform the problem such that IPM becomes applicable
- Even if the transformed problem has more variables, it can be efficiently solved by IPM

Example

- Original problem: n variables, no constraints

$$\min_{\mathbf{x}} \frac{1}{2}\|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2^2 + \lambda\|\mathbf{x}\|_1, \quad \lambda > 0$$

Example

- Original problem: n variables, no constraints

$$\min_{\mathbf{x}} \frac{1}{2}\|\mathbf{A}\mathbf{x}-\mathbf{b}\|_2^2 + \lambda\|\mathbf{x}\|_1, \quad \lambda > 0$$

- Introduce new variable $\mathbf{t} \in \mathbb{R}^n$ and new constraints $|x_i| \leq t_i$:

$$\begin{aligned} &\min_{(\mathbf{x},\mathbf{t})} \frac{1}{2}\|\mathbf{A}\mathbf{x}-\mathbf{b}\|_2^2 + \lambda\mathbf{1}^\top\mathbf{t} \\ \text{s.t.} \quad & -\mathbf{t} \leq \mathbf{x} \leq \mathbf{t} \end{aligned}$$

Example

- Original problem: n variables, no constraints

$$\min_{\mathbf{x}} \frac{1}{2}\|\mathbf{A}\mathbf{x}-\mathbf{b}\|_2^2 + \lambda\|\mathbf{x}\|_1, \quad \lambda > 0$$

- Introduce new variable $\mathbf{t} \in \mathbb{R}^n$ and new constraints $|x_i| \leq t_i$:

$$\begin{aligned} &\min_{(\mathbf{x},\mathbf{t})} \frac{1}{2}\|\mathbf{A}\mathbf{x}-\mathbf{b}\|_2^2 + \lambda\mathbf{1}^\top\mathbf{t} \\ \text{s.t.} \quad & -\mathbf{t} \leq \mathbf{x} \leq \mathbf{t} \end{aligned}$$

- New problem has $2n$ variables and $2n$ constraints, bit it is **smooth**!

Example

- Original problem: n variables, no constraints

$$\min_{\mathbf{x}} \frac{1}{2}\|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2^2 + \lambda\|\mathbf{x}\|_1, \quad \lambda > 0$$

- Introduce new variable $\mathbf{t} \in \mathbb{R}^n$ and new constraints $|x_i| \leq t_i$:

$$\begin{aligned} \min_{(\mathbf{x},\mathbf{t})} \ & \frac{1}{2}\|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2^2 + \lambda\mathbf{1}^\top\mathbf{t} \\ \text{s.t.} \ & -\mathbf{t} \leq \mathbf{x} \leq \mathbf{t} \end{aligned}$$

- New problem has $2n$ variables and $2n$ constraints, bit it is **smooth**!
- **Important point**: problems are equivalent! If you solve one of them, you can derive solution of the other one

Problem transformation and efficient solving

- Given convex problem P_0

Problem transformation and efficient solving

- Given convex problem P_0
- The following sequential transformations are performed

$$P_0 \to P_1 \to \ldots \to P_K,$$

where P_K is a problem that is solved by IPM

Problem transformation and efficient solving

- Given convex problem P_0
- The following sequential transformations are performed

$$P_0 \to P_1 \to \ldots \to P_K,$$

  where P_K is a problem that is solved by IPM
- Efficient solving of P_K

Problem transformation and efficient solving

- Given convex problem P_0
- The following sequential transformations are performed

$$P_0 \to P_1 \to \ldots \to P_K,$$

  where P_K is a problem that is solved by IPM
- Efficient solving of P_K
- Reverse transformation from solution of P_K to solution of P_0

Problem transformation and efficient solving

- Given convex problem P_0
- The following sequential transformations are performed

$$P_0 \to P_1 \to \ldots \to P_K,$$

  where P_K is a problem that is solved by IPM
- Efficient solving of P_K
- Reverse transformation from solution of P_K to solution of P_0
- Problem P_K can have more variables/constraints, but IPM is still extremely efficient even in these cases

Examples of problem transformations

- The rules for convex function transformation lead to the problem transformations!

Examples of problem transformations

- The rules for convex function transformation lead to the problem transformations!
- $\max\{f_1(\mathbf{x}), f_2(\mathbf{x})\}$

Examples of problem transformations

- The rules for convex function transformation lead to the problem transformations!
- $\max\{f_1(\mathbf{x}), f_2(\mathbf{x})\}$
  - Introduce new variable $t = \max\{f_1(\mathbf{x}), f_2(\mathbf{x})\}$

Examples of problem transformations

- The rules for convex function transformation lead to the problem transformations!
- $\max\{f_1(\mathbf{x}), f_2(\mathbf{x})\}$
  - Introduce new variable $t = \max\{f_1(\mathbf{x}), f_2(\mathbf{x})\}$
  - Add constraints $f_1(\mathbf{x}) \leq t,\ f_2(\mathbf{x}) \leq t$

Examples of problem transformations

- The rules for convex function transformation lead to the problem transformations!
- $\max\{f_1(\mathbf{x}), f_2(\mathbf{x})\}$
  - Introduce new variable $t = \max\{f_1(\mathbf{x}), f_2(\mathbf{x})\}$
  - Add constraints $f_1(\mathbf{x}) \le t,\ f_2(\mathbf{x}) \le t$
- $h(f(\mathbf{x}))$

Examples of problem transformations

- The rules for convex function transformation lead to the problem transformations!
- $\max\{f_1(\mathbf{x}), f_2(\mathbf{x})\}$
  - Introduce new variable $t = \max\{f_1(\mathbf{x}), f_2(\mathbf{x})\}$
  - Add constraints $f_1(\mathbf{x}) \leq t,\ f_2(\mathbf{x}) \leq t$
- $h(f(\mathbf{x}))$
  - Introduce new variable $t = f(\mathbf{x})$

Examples of problem transformations

- The rules for convex function transformation lead to the problem transformations!
- $\max\{f_1(\mathbf{x}), f_2(\mathbf{x})\}$
  - Introduce new variable $t = \max\{f_1(\mathbf{x}), f_2(\mathbf{x})\}$
  - Add constraints $f_1(\mathbf{x}) \leq t,\ f_2(\mathbf{x}) \leq t$
- $h(f(\mathbf{x}))$
  - Introduce new variable $t = f(\mathbf{x})$
  - Add constraints $f(\mathbf{x}) \leq t$

From convexity proof to applicability of IPM

$$\min_{\mathbf{x} \in \mathbb{R}^n} f_0(\mathbf{x})$$
$$\text{s.t. } f_i(\mathbf{x}) \leq 0,\ i = 1, \ldots, m$$
$$\mathbf{A}\mathbf{x} = \mathbf{b}$$

- Rules of construction f_i give the proof of convexity

From convexity proof to applicability of IPM

$$
\begin{aligned}
&\min_{\mathbf{x}\in\mathbb{R}^n} f_0(\mathbf{x}) \\
\text{s.t. } & f_i(\mathbf{x}) \le 0,\ i = 1, \dots, m \\
& \mathbf{A}\mathbf{x} = \mathbf{b}
\end{aligned}
$$

- Rules of construction f_i give the proof of convexity
- The same parsing leads to a standard form of problem appropriate for IPM

Disciplined convex programming (DCP)

- Set variables and fixed parameters

Disciplined convex programming (DCP)

- Set variables and fixed parameters
- Objective function and constraints are constructed according to pre-defined rules

Disciplined convex programming (DCP)

- Set variables and fixed parameters
- Objective function and constraints are constructed according to pre-defined rules
- The problem is convex by construction

Disciplined convex programming (DCP)

- Set variables and fixed parameters
- Objective function and constraints are constructed according to pre-defined rules
- The problem is convex by construction
- It is automatically parsed by elements

Disciplined convex programming (DCP)

- Set variables and fixed parameters
- Objective function and constraints are constructed according to pre-defined rules
- The problem is convex by construction
- It is automatically parsed by elements
- It is reduced to the form suitable for IPM

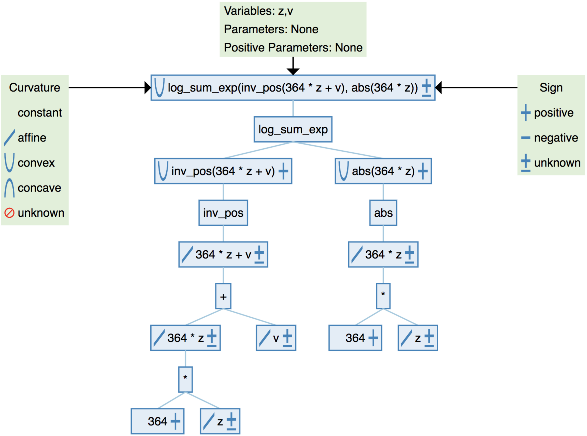
Disciplined convex programming (DCP)

- Set variables and fixed parameters
- Objective function and constraints are constructed according to pre-defined rules
- The problem is convex by construction
- It is automatically parsed by elements
- It is reduced to the form suitable for IPM
- It is solved by some standard IPM solver

Disciplined convex programming (DCP)

- Set variables and fixed parameters
- Objective function and constraints are constructed according to pre-defined rules
- The problem is convex by construction
- It is automatically parsed by elements
- It is reduced to the form suitable for IPM
- It is solved by some standard IPM solver
- The solution of the original problem is reconstructed

Example of parsing the convex function expression and convexity verification



More examples you can find in `http://dcp.stanford.edu/`

Take home message on DCP

Pro:

Take home message on DCP

Pro:

- Convexity verification and generation of the problem transformation for IPM

Take home message on DCP

Pro:

- Convexity verification and generation of the problem transformation for IPM
- Problem generation: elementary convex functions + composition rules and transformations

Take home message on DCP

Pro:

- Convexity verification and generation of the problem transformation for IPM
- Problem generation: elementary convex functions + composition rules and transformations
- It is very similar to standard math notation

Take home message on DCP

Pro:

- Convexity verification and generation of the problem transformation for IPM
- Problem generation: elementary convex functions + composition rules and transformations
- It is very similar to standard math notation

Contra:

Take home message on DCP

Pro:

- Convexity verification and generation of the problem transformation for IPM
- Problem generation: elementary convex functions + composition rules and transformations
- It is very similar to standard math notation

Contra:

- It is not about «plug & play» or «try my code»

Take home message on DCP

Pro:

- Convexity verification and generation of the problem transformation for IPM
- Problem generation: elementary convex functions + composition rules and transformations
- It is very similar to standard math notation

Contra:

- It is not about «plug & play» or «try my code»
- You can not write arbitrary problem and hope that it will be convex

Solvers for the general optimization problems

- ipopt
- Pyomo
- Gurobi

Take home message from this lecture

- Composition rules to construct convex functions

Take home message from this lecture

- Composition rules to construct convex functions
- Reduce problems to some standard forms

Take home message from this lecture

- Composition rules to construct convex functions
- Reduce problems to some standard forms
- Disciplined convex programming

Take home message from this lecture

- Composition rules to construct convex functions
- Reduce problems to some standard forms
- Disciplined convex programming
- Examples

Take home message from this lecture

- Composition rules to construct convex functions
- Reduce problems to some standard forms
- Disciplined convex programming
- Examples
- Other solvers for solving optimization problems