An Introduction to CVX

Amirpasha Shirazinia

Signals and Systems Division
Department of Engineering Sciences
Uppsala University

November 4, 2014
What is CVX?

- CVX is a MATLAB-based software package for solving convex optimization problems.
- **Feature:** CVX is written in a high-level form, i.e., close to mathematical notations.
- 1748 citations (according to google).
What is CVX?

- CVX is a MATLAB–based software package for solving convex optimization problems.

- Feature: CVX is written in a high–level form, i.e., close to mathematical notations.

- 1748 citations (according to google).

- Restriction: can be applied to convex programs,

- What if the problem you are dealing with is non-convex:
  - Try to convexify the non-convex problem,
  - Try to relax the problem in order to make it convex,
  - Go for a sub-optimal locally minimizing approach, e.g., fmincon.
What is CVX?

- CVX is a MATLAB–based software package for solving convex optimization problems.
- **Feature:** CVX is written in a high–level form, i.e., close to mathematical notations.
- 1748 citations (according to google).
- **Restriction:** can be applied to convex programs,
- What if the problem you are dealing with is non-convex:
  - Try to **convexify** the non-convex problem,
  - Try to **relax** the problem in order to make it convex,
  - Go for a **sub-optimal** locally minimizing approach, e.g., fmincon.
- **Download & install CVX.**
Scalar Example

\[
\begin{align*}
\text{minimize} & \quad (x + y + 3)^2 \\
\text{subject to} & \quad y \geq 0 \\
& \quad 0 \leq x \leq 1.
\end{align*}
\] (1)

How to translate Problem (1) into CVX?
Scalar Example

minimize \((x + y + 3)^2\)
subject to \(y \geq 0\)
\(0 \leq x \leq 1.\)  

(1)

How to translate Problem (1) into CVX?

```matlab
>> cvx_begin
```
Scalar Example

minimize \( (x + y + 3)^2 \)
subject to \( y \geq 0 \)
\( 0 \leq x \leq 1 \). \hspace{1cm} (1)

How to translate Problem (1) into CVX?

>> cvx_begin
>> variables x y
Scalar Example

minimize \((x + y + 3)^2\)
subject to \(y \geq 0\)
\(0 \leq x \leq 1.\)  \hspace{1cm} (1)

How to translate Problem (1) into CVX?

```matlab
>> cvx_begin
>> variables x y
>> y >= 0
```
Scalar Example

minimize \((x + y + 3)^2\)
subject to \[y \geq 0\]
\[0 \leq x \leq 1.\]

How to translate Problem (1) into CVX?

```matlab
>> cvx_begin
>> variables x y
>> y >= 0
>> x >= 0
```
Scalar Example

\[
\begin{align*}
\text{minimize} & \quad (x + y + 3)^2 \\
\text{subject to} & \quad y \geq 0 \\
& \quad 0 \leq x \leq 1.
\end{align*}
\] (1)

How to translate Problem (1) into CVX?

```
>> cvx_begin
>> variables x y
>> y >= 0
>> x >= 0
>> x <= 1
```
Scalar Example

\[
\begin{align*}
\text{minimize} \quad & (x + y + 3)^2 \\
\text{subject to} \quad & y \geq 0 \\
& 0 \leq x \leq 1.
\end{align*}
\]  

(1)

How to translate Problem (1) into CVX?

\[
>> \text{cvx_begin} \\
>> \text{variables } x \ y \\
>> y >= 0 \\
>> x >= 0 \\
>> x <= 1 \\
>> \text{minimize}(x + y + 3)^2
\]
Scalar Example

minimize $(x + y + 3)^2$
subject to $y \geq 0$
0 $\leq x \leq 1$.  \hspace{1cm} (1)

How to translate Problem (1) into CVX?

```matlab
>> cvx_begin
>> variables x y
>> y >= 0
>> x >= 0
>> x <= 1
>> minimize((x + y + 3)^2)
>> cvx_end
```
Some Useful Commands & Hints

- `cvx_status`: gives the status of the solution – solved/infeasible/unbounded,
- `cvx_optval`: gives the optimal value,
- `cvx_quiet(true)`: displays off the optimization procedure,
- Affine equality constraints are denoted by `==`,
- Flexibility: CVX commands can be used in MATLAB loops,
- Lots of other hints available in CVX user’s guide.
• Problem 4.8e (Boyd): Finding optimal probabilities for some weighting coefficients $\{\alpha_i\}_{i=1}^N$

minimize \[ \sum_{i=1}^N \alpha_i p_i \]
subject to \[ p_i \geq 0, \ \forall i \]
\[ \sum_{i=1}^N p_i = 1. \]
• Problem 4.8e (Boyd): Finding optimal probabilities for some weighting coefficients \( \{\alpha_i\}_{i=1}^{N} \)

\[
\begin{align*}
\text{minimize} & \quad \sum_{i=1}^{N} \alpha_i p_i \\
\text{subject to} & \quad p_i \geq 0, \quad \forall i \\
& \quad \sum_{i=1}^{N} p_i = 1.
\end{align*}
\]

\[>> N = 10; \text{alpha} = \text{rand}(1,N)\]
Problem 4.8e (Boyd): Finding optimal probabilities for some weighting coefficients \( \{\alpha_i\}_{i=1}^N \)

\[
\text{minimize} \quad \sum_{i=1}^{N} \alpha_i p_i \\
\text{subject to} \quad p_i \geq 0, \; \forall i \\
\sum_{i=1}^{N} p_i = 1.
\]  

\[ \text{(2)} \]

\[ \text{>> N = 10; alpha = rand(1, N)} \]
\[ \text{>> cvx_begin} \]
Problem 4.8e (Boyd): Finding optimal probabilities for some weighting coefficients \( \{\alpha_i\}_{i=1}^N \)

\[
\begin{align*}
\text{minimize} & \quad \sum_{i=1}^{N} \alpha_i p_i \\
\text{subject to} & \quad p_i \geq 0, \quad \forall i \\
& \quad \sum_{i=1}^{N} p_i = 1.
\end{align*}
\] (2)

\[\begin{align*}
\text{	exttt{>> N = 10; alpha = rand(1,N)}} \\
\text{	exttt{>> cvx_begin}} \\
\text{	exttt{>> variable p(N)}}
\end{align*}\]
Problem 4.8e (Boyd): Finding optimal probabilities for some weighting coefficients \( \{\alpha_i\}_{i=1}^N \)

\[
\begin{align*}
\text{minimize} & \quad \sum_{i=1}^{N} \alpha_i p_i \\
\text{subject to} & \quad p_i \geq 0, \quad \forall i \\
& \quad \sum_{i=1}^{N} p_i = 1.
\end{align*}
\]

\( \quad \text{>> N = 10; alpha = rand(1, N)} \)
\( \quad \text{>> cvx_begin} \)
\( \quad \text{>> variable p(N)} \)
\( \quad \text{>> p >= 0} \)
Problem 4.8e (Boyd): Finding optimal probabilities for some weighting coefficients \( \{ \alpha_i \}_{i=1}^N \)

\[
\begin{align*}
\text{minimize} & \quad \sum_{i=1}^{N} \alpha_i p_i \\
\text{subject to} & \quad p_i \geq 0, \quad \forall i \\
& \quad \sum_{i=1}^{N} p_i = 1.
\end{align*}
\] (2)

\[
\begin{align*}
\gg\gg N &= 10; \text{alpha} = \text{rand}(1,N) \\
\gg\gg \text{cvx}_\text{begin} \\
\gg\gg \text{variable p}(N) \\
\gg\gg p &= 0 \\
\gg\gg \text{sum}(p) &= 1
\end{align*}
\]
Problem 4.8e (Boyd): Finding optimal probabilities for some weighting coefficients \( \{\alpha_i\}_{i=1}^N \)

\[
\begin{align*}
\text{minimize} & \quad \sum_{i=1}^{N} \alpha_i p_i \\
\text{subject to} & \quad p_i \geq 0, \quad \forall i \\
& \quad \sum_{i=1}^{N} p_i = 1.
\end{align*}
\]

\[
(2)
\]

\[
>> \text{N = 10; alpha = rand(1,N)} \\
>> \text{cvx_begin} \\
>> \text{variable p(N)} \\
>> \text{p >= 0} \\
>> \text{sum(p) == 1} \\
>> \text{minimize(alpha*p)}
\]
Problem 4.8e (Boyd): Finding optimal probabilities for some weighting coefficients \( \{\alpha_i\}_{i=1}^N \)

\[
\begin{align*}
\text{minimize} & \quad \sum_{i=1}^{N} \alpha_i p_i \\
\text{subject to} & \quad p_i \geq 0, \ \forall i \\
& \quad \sum_{i=1}^{N} p_i = 1.
\end{align*}
\]

\( N = 10; \ \alpha = \text{rand}(1, N) \)

\[
\begin{align*}
\text{cvx_begin} \\
\text{variable} \ p(\ N) \\
p \ \geq \ 0 \\
\text{sum}(p) \ = \ 1 \\
\text{minimize} \ (\alpha \times p) \\
\text{cvx_end}
\end{align*}
\]
Problem 4.26 (Boyd):

\[
\begin{align*}
\text{minimize} & \quad x^\top x + y + z \\
\text{subject to} & \quad x^\top x \leq yz \\
& \quad y \geq 0, \quad z \geq 0
\end{align*}
\]

Question 1: Is the objective function convex?
• Problem 4.26 (Boyd):

\[
\begin{align*}
\text{minimize} & \quad \begin{pmatrix} x & y & z \end{pmatrix}^T \begin{pmatrix} x & y & z \end{pmatrix} + y + z \\
\text{subject to} & \quad \begin{pmatrix} x & y & z \end{pmatrix}^T \begin{pmatrix} x & y & z \end{pmatrix} \leq yz \\
& \quad y \geq 0, \quad z \geq 0
\end{align*}
\]

(3)

• Question 1: Is the objective function convex? Yes! 😊

• Question 2: Are the constraints convex sets?
Vector Example

- Problem 4.26 (Boyd):

\[
\begin{align*}
\text{minimize} & \quad x^\top x + y + z \\
\text{subject to} & \quad x^\top x \leq yz \\
& \quad y \geq 0, \ z \geq 0
\end{align*}
\]

(3)

- Question 1: Is the objective function convex? Yes! 😊
- Question 2: Are the constraints convex sets? Let’s call CVX first 📱

```matlab
>> N = 10;
>> cvx_begin
>> variables x(N, 1) y z
>> y >= 0
>> z >= 0
>> x' * x - y * z <= 0
```
Problem 4.26 (Boyd):

\[
\begin{align*}
\text{minimize} & \quad x^\top x + y + z \\
\text{subject to} & \quad x^\top x \leq yz \\
& \quad y \geq 0, \quad z \geq 0
\end{align*}
\]  

(3)

Question 1: Is the objective function convex? **Yes! 😊**

Question 2: Are the constraints convex sets? Let’s call CVX first 🤔

```matlab
>> N = 10;
>> cvx_begin
>> variables x(N,1) y z
>> y >= 0
>> z >= 0
>> x' * x - y * z <= 0

Disciplined convex programming error:
```
Vector Example (Cont’d)

- Why the constraint $x^\top x \leq yz$ does not form a convex region?
Vector Example (Cont’d)

- Why the constraint $x^\top x \leq yz$ does not form a convex region?
  - convex $\leq$ ???
• Why the constraint $x^\top x \leq yz$ does not form a convex region?
  • convex $\leq ???$
  • ??? should be an affine or concave function.
Vector Example (Cont’d)

- Why the constraint $\mathbf{x}^\top \mathbf{x} \leq yz$ does not form a convex region?
  - convex $\leq$ ???
  - ?? should be an affine or concave function.
  - Now, via the Hessian matrix, check if $f(y, z) = yz$ is concave (Problem 3.16 Boyd).
Vector Example (Cont’d)

- Why the constraint $\mathbf{x}^\top \mathbf{x} \leq yz$ does not form a convex region?
  - convex $\leq ???$
  - ??? should be an affine or concave function.
  - Now, via the Hessian matrix, check if $f(y, z) = yz$ is concave (Problem 3.16 Boyd).
- How to convexify Problem (2)?

\[
\mathbf{x}^\top \mathbf{x} \leq yz \iff \left\| \begin{bmatrix} 2x \\ y - z \end{bmatrix} \right\|_2 \leq y + z.
\]  (4)
Vector Example (Cont’d)

- Why the constraint $\mathbf{x}^\top \mathbf{x} \leq yz$ does not form a convex region?
  - convex $\leq ???$
  - ??? should be an affine or concave function.
  - Now, via the Hessian matrix, check if $f(y, z) = yz$ is concave (Problem 3.16 Boyd).

- How to convexify Problem (2)?

  $$
  \mathbf{x}^\top \mathbf{x} \leq yz \iff \left\| \begin{bmatrix} 2x \\ y - z \end{bmatrix} \right\|_2 \leq y + z. 
  \tag{4}
  $$

- Now, we have a convex problem 😊

  $$
  \begin{align*}
  \text{minimize} & \quad \mathbf{x}^\top \mathbf{x} + y + z \\
  \text{subject to} & \quad \left\| \begin{bmatrix} 2x \\ y - z \end{bmatrix} \right\|_2 \leq y + z \\
  & \quad y \geq 0, \ z \geq 0
  \end{align*}
  \tag{5}
  $$
Vector Example (Cont’d)

• CVX code:
  >>> N = 10;
  >>> cvx_begin
  >>> variables x(N,1) y z
  >>> y >= 0
  >>> z >= 0
  >>> norm([2x; y - z], 2) <= y + z
  >>> minimize(x' * x + y + z)
  >>> cvx_end
Matrix Example: Sensing/Measurement Matrix design in a Bayesian Framework

- Consider the linear Gaussian model \( y = Ax + n \),
- where \( x \sim \mathcal{N}(0, R) \), \( n \sim \mathcal{N}(0, \sigma^2 I) \), such that \( x, y, n \in \mathbb{R}^N \) and \( A \in \mathbb{R}^{N \times N} \).
- The MMSE estimator of \( x \) given the noisy measurements \( y \) becomes

  \[
  \hat{x}_{mmse} = \arg \min_{\hat{x}} \mathbb{E}[\|x - \hat{x}\|_2^2] = \frac{1}{\sigma^2} \left( R^{-1} + \frac{1}{\sigma^2} A^\top A \right)^{-1} A^\top y
  \]

- The corresponding MSE, to be minimized, is

  \[
  \text{MSE} = \text{Tr} \left\{ \left( R^{-1} + \frac{1}{\sigma^2} A^\top A \right)^{-1} \right\}
  \] (6)
Matrix Example: Sensing/Measurement Matrix design in a Bayesian Framework

- Consider the linear Gaussian model $y = Ax + n$,
- where $x \sim \mathcal{N}(0, R)$, $n \sim \mathcal{N}(0, \sigma^2 I)$, such that $x, y, n \in \mathbb{R}^N$ and $A \in \mathbb{R}^{N \times N}$.
- The MMSE estimator of $x$ given the noisy measurements $y$ becomes

$$
\hat{x}_{mmse} = \arg \min_{\hat{x}} \mathbb{E}[\|x - \hat{x}\|_2^2] = \frac{1}{\sigma^2} \left( R^{-1} + \frac{1}{\sigma^2} A^\top A \right)^{-1} A^\top y
$$

- The corresponding MSE, to be minimized, is

$$
\text{MSE} = \text{Tr} \left\{ \left( R^{-1} + \frac{1}{\sigma^2} A^\top A \right)^{-1} \right\} \quad (6)
$$

- Assume the total available power $P > 0$, we have

$$
\mathbb{E}[\|Ax\|_2^2] = \ldots = \text{Tr} \{ ARA^\top \} \leq P \quad (7)
$$
The sensing/measurement matrix design problem is posed as

\[
\begin{align*}
\text{minimize} & \quad \text{Tr} \left\{ \left( R^{-1} + \frac{1}{\sigma^2} A^\top A \right)^{-1} \right\} \\
\text{subject to} & \quad \text{Tr} \left\{ RA^\top A \right\} \leq P
\end{align*}
\]
• The sensing/measurement matrix design problem is posed as

\[
\begin{align*}
\text{minimize} & \quad \text{Tr} \left\{ (\mathbf{R}^{-1} + \frac{1}{\sigma^2} \mathbf{A}^\top \mathbf{A})^{-1} \right\} \\
\text{subject to} & \quad \text{Tr} \left\{ \mathbf{R} \mathbf{A}^\top \mathbf{A} \right\} \leq P
\end{align*}
\]  

(8)

• Problem (8) is not convex in \( \mathbf{A} \), but convex in \( \mathbf{A}^\top \mathbf{A} \).

• Let \( \mathbf{X} \triangleq \mathbf{A}^\top \mathbf{A} \), we have an equivalent optimization problem

\[
\begin{align*}
\text{minimize} & \quad \text{Tr} \left\{ (\mathbf{R}^{-1} + \frac{1}{\sigma^2} \mathbf{X})^{-1} \right\} \\
\text{subject to} & \quad \text{Tr} \left\{ \mathbf{R} \mathbf{X} \right\} \leq P \\
& \quad \mathbf{X} \succeq 0
\end{align*}
\]  

(9)
Let’s translate the optimization problem (8) into CVX codes

```matlab
>> N = 3; rho = 0.5; P = 10; sigma_sq = 0.01;
>> R = [1 rho rho^2; rho 1 rho; rho^2 rho 1];
>> cvx_begin sdp
>> variable X(N,N) symmetric
>> X == semidefinite(N,N)
>> trace(R*X) <= P
>> minimize(trace_inv(R^(-1) + X/sigma_sq))
```
Let’s translate the optimization problem (8) into CVX codes

```matlab
>> N = 3; rho = 0.5; P = 10; sigma_sq = 0.01;
>> R = [1 rho rho^2; rho 1 rho ; rho^2 rho 1];
>> cvx_begin sdp
>> variable X(N,N) symmetric
>> X == semidefinite(N,N)
>> trace(R * X) <= P
>> minimize(trace_inv(R^(-1) + X/sigma_sq))
>> cvx_end
```

How to get back $A$ from $X$?

- For example, use the cholesky factorization; in MATLAB $A = \text{chol}(X)$. 
A Little into Compressed Sensing (CS)

- CS answers the following question:
  - In a linear set of equations $y = Ax$, would it be possible to uniquely solve the equations if the number of unknowns, i.e. $\dim(x)$, is larger than the number of equations, i.e. $\dim(y)$?
A Little into Compressed Sensing (CS)

- CS answers the following question:
  - In a linear set of equations \( y = Ax \), would it be possible to uniquely solve the equations if the number of unknowns, i.e. \( \dim(x) \), is larger than the number of equations, i.e. \( \dim(y) \)?

- Yes! But under the condition that the unknown variables are known to be sparse \((K \ll N)\) plus some mild conditions.
- We know that the source, \( x \), is \textit{sparse}. 
We know that the source, $x$, is sparse.
We observe the linear measurements $y = Ax$. 
Optimization framework for CS

- We know that the source, \( x \), is \textit{sparse}.
- We observe the \textit{linear measurements} \( y = Ax \).
- \textbf{Goal}: Go for the sparsest solution.

\[
\begin{align*}
\text{minimize} & \quad \|x\|_0 \\
\text{subject to} & \quad y = Ax 
\end{align*}
\]  

where the \( \ell_0 \)-norm \( \|x\|_0 \triangleq \text{card}(x) = \#\text{non-zero components} \).
- Check whether or not the optimization problem (10) is convex ...
  - \textbf{Hint}: Check the geometry of \( \ell_0 \) ...
Optimization framework for CS

- We know that the source, $x$, is sparse.
- We observe the linear measurements $y = Ax$.
- **Goal:** Go for the sparsest solution.

\[
\begin{align*}
\text{minimize} & \quad \|x\|_0 \\
\text{subject to} & \quad y = Ax
\end{align*}
\]  

(10)

- where the $\ell_0$–norm $\|x\|_0 \triangleq \text{card}(x) = \#\text{non-zero components}$.
- Check whether or not the optimization problem (10) is convex ...
  - **Hint:** Check the geometry of $\ell_0$ ...
  - **NOT CONVEX!**
• How to tackle the problem?

• **Standard approach:** Relax the objective function to the nearest convex norm, i.e., $\ell_1$-norm, a.k.a. basis pursuit (BP),

$$\text{minimize} \quad \|x\|_1$$

subject to \quad $y = Ax$ \quad \text{(11)}

• where $\|x\|_1 = \sum_{n=1}^{N} |x_n|$. 
- How to tackle the problem?
- **Standard approach:** Relax the objective function to the nearest convex norm, i.e., $\ell_1$-norm, a.k.a. basis pursuit (BP),

\[
\begin{align*}
\text{minimize} & \quad \|x\|_1 \\
\text{subject to} & \quad y = Ax
\end{align*}
\]

(11)

where $\|x\|_1 = \sum_{n=1}^{N} |x_n|$.

- The **miracle** happens: Under some conditions, the solutions to (11) and (10) are the same. CS theory also determines how many measurements need to be acquired in order for this equivalence to hold.
Optimization framework for CS: Example

\[
\begin{align*}
\text{>> } N &= 50; \text{ M } = 25; \text{ K } = 5; \\
\text{>> supp } &= \text{ randsample}(N,K); \text{ x } = \text{ zeros}(N,1); \text{ x(supp) } = \text{ randn}(K,1); \\
\text{>> A } &= \text{ randn}(M,N); \text{ s } = \text{ sqrt}(\text{sum}(A.^2)); \text{ S } = \text{ diag}(1./s); \\
\text{>> A } &= \text{ A } \times \text{ S}; \\
\text{>> y } &= \text{ A } \times \text{ x}; \\
\text{>> cvx_begin} \\
\text{>> variable } \text{x_hat}(N,1) \\
\text{>> y } &= \text{ A } \times \text{x_hat} \\
\text{>> minimize(} \text{norm(x_hat,1)} \text{)} \\
\text{>> cvx_end} \\
\text{>> plot(1:N,x,1:N,'s',x_hat,'o')} 
\end{align*}
\]
• CVX is cool!
References