EE364a Homework 6 solutions

4.40 LPs, QPs, QCQPs, and SOCPs as SDPs. Express the following problems as SDPs.

(a) The LP (4.27).

Solution.

\[
\begin{align*}
\text{minimize} & \quad c^T x + d \\
\text{subject to} & \quad \text{diag}(Gx - h) \preceq 0 \\
& \quad Ax = b.
\end{align*}
\]

(b) The QP (4.34), the QCQP (4.35) and the SOCP (4.36). Hint. Suppose \( A \in S^n_{++} \), \( C \in S^s \), and \( B \in \mathbb{R}^{r \times s} \). Then

\[
\begin{bmatrix}
A & B \\
B^T & C
\end{bmatrix} \succeq 0 \iff C - B^T A^{-1} B \succeq 0.
\]

For a more complete statement, which applies also to singular \( A \), and a proof, see §A.5.5.

Solution.

i. QP. Express \( P = WW^T \) with \( W \in \mathbb{R}^{n \times r} \).

\[
\begin{align*}
\text{minimize} & \quad t + 2q^T x + r \\
\text{subject to} & \quad \begin{bmatrix} I & W^T x \\ x^T W & tI \end{bmatrix} \succeq 0 \\
& \quad \text{diag}(Gx - h) \preceq 0 \\
& \quad Ax = b,
\end{align*}
\]

with variables \( x, t \in \mathbb{R} \).

ii. QCQP. Express \( P_i = W_i W_i^T \) with \( W_i \in \mathbb{R}^{n \times r_i} \).

\[
\begin{align*}
\text{minimize} & \quad t_0 + 2q_0^T x + r_0 \\
\text{subject to} & \quad t_i + 2q_i^T x + r_i \preceq 0, \quad i = 1, \ldots, m \\
& \quad \begin{bmatrix} I & W_i^T x \\ x^T W_i & t_i I \end{bmatrix} \succeq 0, \quad i = 0, 1, \ldots, m \\
& \quad Ax = b,
\end{align*}
\]

with variables \( x, t_i \in \mathbb{R} \).

iii. SOCP.

\[
\begin{align*}
\text{minimize} & \quad c^T x \\
\text{subject to} & \quad \begin{bmatrix} (c_i^T x + d_i) I & A_i x + b_i \\
(A_i x + b_i)^T & (c_i^T x + d_i) I \end{bmatrix} \succeq 0, \quad i = 1, \ldots, N \\
& \quad Fx = g.
\end{align*}
\]
By the result in the hint, the constraint is equivalent with \( \|A_i x + b_i\|_2 < c_i^T x + d_i \) when \( c_i^T x + d_i > 0 \). We have to check the case \( c_i^T x + d_i = 0 \) separately. In this case, the LMI constraint means \( A_i x + b_i = 0 \), so we can conclude that the LMI constraint and the SOC constraint are equivalent.

7.1 Linear measurements with exponentially distributed noise. Show how to solve the ML estimation problem (7.2) when the noise is exponentially distributed, with density

\[
p(z) = \begin{cases} (1/a)e^{-z/a} & z \geq 0 \\ 0 & z < 0, \end{cases}
\]

where \( a > 0 \).

**Solution.** Solve the LP

\[
\begin{align*}
\text{minimize} & \quad 1^T(y - Ax) \\
\text{subject to} & \quad Ax \preceq y.
\end{align*}
\]

8.3 Euclidean projection on proper cones.

i. Nonnegative orthant. Show that Euclidean projection onto the nonnegative orthant is given by the expression on page 399.

**Solution.** The inner product of two nonnegative vectors is zero if and only the componentwise product is zero. We can therefore solve the equations

\[
x_{0,i} = x_{+,i} - x_{-,i}, \quad x_{+,i} \geq 0, \quad x_{-,i} \geq 0, \quad x_{+,i}x_{-,i} = 0,
\]

for \( i = 1, \ldots, n \). If \( x_{0,i} > 0 \) the solution is \( x_{+,i} = x_{0,i}, x_{-,i} = 0 \). If \( x_{0,i} < 0 \) the solution is \( x_{+,i} = 0, x_{-,i} = -x_{0,i} \). If \( x_{0,i} = 0 \) the solution is \( x_{+,i} = x_{-,i} = 0 \).

ii. Positive semidefinite cone. Show that Euclidean projection onto the positive semidefinite cone is given by the expression on page ??.

**Solution.** Define \( \tilde{X}_+ = V^T X_+ V, \tilde{X}_- = V^T X_- V \). These matrices must satisfy

\[
\Lambda = \tilde{X}_+ - \tilde{X}_-, \quad \tilde{X}_+ \succeq 0, \quad \tilde{X}_- \succeq 0, \quad \text{tr}(\tilde{X}_+\tilde{X}_-) = 0.
\]

The first condition implies that the off-diagonal elements are equal: \( (\tilde{X}_+)^{ij} = (\tilde{X}_-)^{ij} \) if \( i \neq j \). The third equation implies

\[
\text{tr}(\tilde{X}_+X_-) = \sum_{i=1}^n (\tilde{X}_+)^{ii}(\tilde{X}_-)^{ii} + \sum_{i=1}^n \sum_{j \neq i} (\tilde{X}_+)^{ij}(\tilde{X}_-)^{ij} = 0
\]

which is only possible if

\[
(\tilde{X}_+)^{ij} = (\tilde{X}_-)^{ij} = 0, \quad i \neq j
\]

and

\[
(\tilde{X}_+)^{ii}(\tilde{X}_-)^{ii} = 0, \quad i = 1, \ldots, n.
\]
In other words, $\tilde{X}_+$ and $\tilde{X}_-$ are diagonal, with a complementary zero-nonzero pattern on the diagonal, \textit{i.e.},

\[(\tilde{X}_+)_{ii} = \max\{\lambda_i, 0\}, \quad (\tilde{X}_0)_{ii} = \max\{-\lambda_i, 0\}.\]

iii. \textit{Second-order cone.} Show that the Euclidean projection of $(x_0, t_0)$ on the second-order cone

\[K = \{(x, t) \in \mathbb{R}^{n+1} \mid \|x\|_2 \leq t\}\]

is given by

\[P_K(x_0, t_0) = \begin{cases} 
0 & \|x_0\|_2 \leq -t_0 \\
(x_0, t_0) & \|x_0\|_2 \leq t_0 \\
(1/2)(1 + t_0/\|x_0\|_2)(x_0, \|x_0\|_2) & \|x_0\|_2 \geq |t_0|.
\end{cases}\]

\textbf{Solution.} The second-order cone is self-dual, so the conditions are

\[x_0 = u - v, \quad t_0 = \mu - \tau, \quad \|u\|_2 \leq \mu, \quad \|v\|_2 \leq \tau, \quad u^Tv + \mu\tau = 0.\]

It follows from the Cauchy-Schwarz inequality that the last three conditions are satisfied if one of the following three cases holds.

- $\mu = 0, u = 0, \|v\|_2 \leq \tau$. The first two conditions give $v = -x_0, t_0 = -\tau$. The fourth condition implies $t_0 \leq 0$, and $\|x_0\|_2 \leq -t_0$.
  In this case $(x_0, t_0)$ is in the negative second-order cone, and its projection is the origin.

- $\tau = 0, v = 0, \|u\|_2 \leq \mu$. The first two conditions give $u = x_0, \mu = t_0$. The third condition implies $\|x_0\|_2 \leq t_0$.
  In this case $(x_0, t_0)$ is in the second-order cone, so it is its own projection.

- $\|u\|_2 = \mu > 0, \|v\|_2 = \tau > 0, \tau u = -\mu v$. We can express $v$ as $v = -(\tau/\mu)u$. From $x_0 = u - v$,
  \[x_0 = (1 + \tau/\mu)u, \quad \mu = \|u\|_2,\]
  and therefore $\mu + \tau = \|x_0\|_2$. Also, $t_0 = \mu - \tau$. Solving for $\mu$ and $\tau$ gives
  \[\mu = (1/2)(t_0 + \|x_0\|_2), \quad \tau = (1/2)(-t_0 + \|x_0\|_2).\]

$\tau$ is only positive if $t_0 < \|x_0\|_2$. We obtain

\[u = \frac{t_0 + \|x_0\|_2}{2\|x_0\|_2}x_0, \quad \mu = \frac{\|x_0\|_2 + t_0}{2}, \quad v = \frac{t_0 - \|x_0\|_2}{2\|x_0\|_2}x_0, \quad \tau = \frac{\|x_0\|_2 - t_0}{2}.\]

8.20 \textit{Ellipsoidal approximation from analytic center of linear matrix inequality.} Let $C$ be the solution set of the LMI

\[x_1A_1 + x_2A_2 + \cdots + x_nA_n \preceq B,\]
where $A_i, B \in S^m$, and let $x_{ac}$ be its analytic center. Show that

$$\mathcal{E}_{inner} \subseteq C \subseteq \mathcal{E}_{outer},$$

where

$$\mathcal{E}_{inner} = \{ x \mid (x - x_{ac})^T H(x - x_{ac}) \leq 1 \},$$

$$\mathcal{E}_{outer} = \{ x \mid (x - x_{ac})^T H(x - x_{ac}) \leq m(m-1) \},$$

and $H$ is the Hessian of the logarithmic barrier function

$$-\log \det(B - x_1A_1 - x_2A_2 - \cdots - x_nA_n)$$

evaluated at $x_{ac}$.

**Solution.** Define $F(x) = B - \sum_i x_i A_i$, and $F_{ac} = F(x_{ac})$. The Hessian is given by

$$H_{ij} = \text{tr}(F^{-1}_{ac}A_i F^{-1}_{ac}A_j),$$

so we have

$$(x - x_{ac})^T H(x - x_{ac}) = \sum_{i,j} (x_i - x_{ac,i})(x_j - x_{ac,j}) \text{tr}(F^{-1}_{ac}A_i F^{-1}_{ac}A_j)$$

$$= \text{tr} \left( F^{-1}_{ac}(F(x) - F_{ac}) F^{-1}_{ac}(F(x) - F_{ac}) \right)$$

$$= \text{tr} \left( F^{-1/2}_{ac}(F(x) - F_{ac}) F^{-1/2}_{ac} \right)^2.$$

We first consider the inner ellipsoid. Suppose $x \in \mathcal{E}_{inner}$, i.e.,

$$\text{tr} \left( F^{-1/2}_{ac}(F(x) - F_{ac}) F^{-1/2}_{ac} \right)^2 = \left\| F^{-1/2}_{ac}(F(x) - F_{ac}) F^{-1/2}_{ac} - I \right\|_F^2 \leq 1.$$

This implies that

$$-1 \leq \lambda_i(F^{-1/2}_{ac}(F(x) - F_{ac}) F^{-1/2}_{ac}) - 1 \leq 1,$$

i.e.,

$$0 \leq \lambda_i(F^{-1/2}_{ac}(F(x) - F_{ac}) F^{-1/2}_{ac}) \leq 2$$

for $i = 1, \ldots, m$. In particular, $F(x) \succeq 0$, i.e., $x \in C$.

To prove that $C \subseteq \mathcal{E}_{outer}$, we first note that the gradient of the logarithmic barrier function vanishes at $x_{ac}$, and therefore,

$$\text{tr}(F^{-1}_{ac}A_i) = 0, \quad i = 1, \ldots, n,$$

and therefore

$$\text{tr} \left( F^{-1}_{ac}(F(x) - F_{ac}) \right) = 0, \quad \text{tr} \left( F^{-1}_{ac}F(x) \right) = m.$$
Now assume \( x \in C \). Then
\[
(x - x_{ac})^T H (x - ac) \\
= \text{tr} \left( F^{-1/2}_ac (F(x) - F_ac) F^{-1/2}_ac \right)^2 \\
= \text{tr} \left( F^{-1}_ac (F(x) - F_ac) F^{-1}_ac (F(x) - F_ac) \right) \\
= \text{tr} (F^{-1}_ac F(x) F^{-1}_ac F(x)) - 2 \text{tr} (F^{-1}_ac F(x)) + \text{tr} (F^{-1}_ac F_ac F^{-1}_ac F_ac) \\
= \text{tr} (F^{-1/2}_ac F(x) F^{-1/2}_ac)^2 - m \\
\leq \left( \text{tr}(F^{-1/2}_ac F(x) F^{-1/2}_ac) \right)^2 - m \\
= m^2 - m.
\]

The inequality follows by applying the inequality \( \sum_i \lambda_i^2 \leq (\sum_i \lambda_i)^2 \) for \( \lambda \geq 0 \) to the eigenvalues of \( F^{-1/2}_ac F(x) F^{-1/2}_ac \).

A6.11 **Minimax linear fitting.** Consider a linear measurement model \( y = Ax + v \), where \( x \in \mathbb{R}^n \) is a vector of parameters to be estimated, \( y \in \mathbb{R}^m \) is a vector of measurements, \( v \in \mathbb{R}^m \) is a set of measurement errors, and \( A \in \mathbb{R}^{m \times n} \) with rank \( n \), with \( m \geq n \). We know \( y \) and \( A \), but we don’t know \( v \); our goal is to estimate \( x \). We make only one assumption about the measurement error \( v: \|v\|_\infty \leq \epsilon \).

We will estimate \( x \) using a linear estimator \( \hat{x} = By \); we must choose the estimation matrix \( B \in \mathbb{R}^{n \times m} \). The estimation error is \( e = \hat{x} - x \). We will choose \( B \) to minimize the maximum possible value of \( \|e\|_\infty \), where the maximum is over all values of \( x \) and all values of \( v \) satisfying \( \|v\|_\infty \leq \epsilon \).

i. Show how to find \( B \) via convex optimization.

ii. **Numerical example.** Solve the problem instance given in \texttt{minimax_fit_data.m}.

Display the \( \hat{x} \) you obtain and report \( \|\hat{x} - x^{\text{true}}\|_\infty \). Here \( x^{\text{true}} \) is the value of \( x \) used to generate the measurement \( y \); it is given in the data file.

**Solution.**

i. The problem is to minimize the objective \( f(B) \), where
\[
f(B) = \max_{x, \|v\|_\infty \leq \epsilon} \|\hat{x} - x\|_\infty = \max_{x, \|v\|_\infty \leq \epsilon} \|(BA - I)x + Bv\|_\infty.
\]

The maximum over \( x \) is \( +\infty \), unless we have \( BA = I \), so this will be a constraint for us. (By the way, \( BA = I \) means that \( B \) is a left inverse of \( A \). The associated estimator is called an unbiased linear estimator, since without noise there is no estimation error.) Assuming \( BA = I \), we have
\[
f(B) = \max_{\|v\|_\infty \leq \epsilon} \|Bv\|_\infty = \max_{\|v\|_\infty \leq \epsilon} \max_{i=1,...,m} |b_i^T v| = \epsilon \max_{i=1,...,m} \|b_i\|_1
\]

5
where $b_i^T$ are the rows of $B$. (In fact, $f(B)$ is a norm of $B$ called, for obvious reasons, the max-row-sum norm, and denoted $\|B\|_\infty$.) The problem is thus

\begin{equation}
\begin{aligned}
\text{minimize} & \quad \max_{i=1,\ldots,m} \|b_i\|_1 \\
\text{subject to} & \quad BA = I.
\end{aligned}
\end{equation}

This is evidently a convex problem, with variable $B$. In fact, it is separable in the rows; we can solve for each row separately, by solving

\begin{equation}
\begin{aligned}
\text{minimize} & \quad \|b_i\|_1 \\
\text{subject to} & \quad b_i^TA = e_i^T,
\end{aligned}
\end{equation}

for $i = 1,\ldots,n$.

ii. The solution is given on the following page.

```plaintext
%% minimax linear fitting
minimax_fit_data;

cvx_begin
variable B(n,m)
minimize (norm(B,inf))
subject to
B*A == eye(n);
cvx_end

x = B*y;
fprintf(1, 'estimation error = %f
', norm(x - x_true, inf));
```

A7.7 Thickest slab separating two sets. We are given two sets in $\mathbb{R}^n$: a polyhedron $C_1 = \{x \mid Cx \leq d\}$, defined by a matrix $C \in \mathbb{R}^{m \times n}$ and a vector $d \in \mathbb{R}^m$, and an ellipsoid

\begin{equation}
C_2 = \{Pu + q \mid \|u\|_2 \leq 1\},
\end{equation}

defined by a matrix $P \in \mathbb{R}^{n \times n}$ and a vector $q \in \mathbb{R}^n$. We assume that the sets are nonempty and that they do not intersect. We are interested in the optimization problem

\begin{equation}
\begin{aligned}
\text{maximize} & \quad \inf_{x \in C_1} a^Tx - \sup_{x \in C_2} a^Tx \\
\text{subject to} & \quad \|a\|_2 = 1.
\end{aligned}
\end{equation}

with variable $a \in \mathbb{R}^n$.

Explain how you would solve this problem. You can answer the question by reducing the problem to a standard problem class (LP, QP, SOCP, SDP, ...), or by describing an algorithm to solve it.
Remark. The geometrical interpretation is as follows. If we choose
\[ b = \frac{1}{2} \left( \inf_{x \in C_1} a^T x + \sup_{x \in C_2} a^T x \right), \]
then the hyperplane \( H = \{ x \mid a^T x = b \} \) is the maximum margin separating hyperplane separating \( C_1 \) and \( C_2 \). Alternatively, \( a \) gives us the thickest slab that separates the two sets.

Solution.
\[
\begin{align*}
\text{maximize} & \quad -d^T z - \|P^T a\|_2 - q^T a \\
\text{subject to} & \quad \|a\|_2 \leq 1 \\
& \quad C^T z + a = 0 \\
& \quad z \succeq 0.
\end{align*}
\]

An SOCP.

A12.13 Bandlimited signal recovery from zero-crossings. Let \( y \in \mathbb{R}^n \) denote a bandlimited signal, which means that it can be expressed as a linear combination of sinusoids with frequencies in a band:
\[
y_t = \sum_{j=1}^{B} a_j \cos \left( \frac{2\pi}{n}(f_{\min} + j - 1)t \right) + b_j \sin \left( \frac{2\pi}{n}(f_{\min} + j - 1)t \right), \quad t = 1, \ldots, n,
\]
where \( f_{\min} \) is lowest frequency in the band, \( B \) is the bandwidth, and \( a, b \in \mathbb{R}^B \) are the cosine and sine coefficients, respectively. We are given \( f_{\min} \) and \( B \), but not the coefficients \( a, b \) or the signal \( y \).

We do not know \( y \), but we are given its sign \( s = \text{sign}(y) \), where \( s_t = 1 \) if \( y_t \geq 0 \) and \( s_t = -1 \) if \( y_t < 0 \). (Up to a change of overall sign, this is the same as knowing the ‘zero-crossings’ of the signal, i.e., when it changes sign. Hence the name of this problem.)

We seek an estimate \( \hat{y} \) of \( y \) that is consistent with the bandlimited assumption and the given signs. Of course we cannot distinguish \( y \) and \( \alpha y \), where \( \alpha > 0 \), since both of these signals have the same sign pattern. Thus, we can only estimate \( y \) up to a positive scale factor. To normalize \( \hat{y} \), we will require that \( \|\hat{y}\|_1 = n \), i.e., the average value of \( |y_t| \) is one. Among all \( \hat{y} \) that are consistent with the bandlimited assumption, the given signs, and the normalization, we choose the one that minimizes \( \|\hat{y}\|_2 \).

i. Show how to find \( \hat{y} \) using convex or quasiconvex optimization.

ii. Apply your method to the problem instance with data in \texttt{zero_crossings_data.*}.

The data files also include the true signal \( y \) (which of course you cannot use to find \( \hat{y} \)). Plot \( \hat{y} \) and \( y \), and report the relative recovery error, \( \|y - \hat{y}\|_2 / \|y\|_2 \).

Give one short sentence commenting on the quality of the recovery.

Solution.
i. We can express our estimate as \( \hat{y} = Ax \), where \( x = (a, b) \in \mathbb{R}^{2B} \) is the vector of cosine and sinusoid coefficients, and we define the matrix

\[
A = [C \ S] \in \mathbb{R}^{n \times 2B},
\]

where \( C, S \in \mathbb{R}^{n \times B} \) have entries

\[
C_{tj} = \cos(2\pi(f_{\text{min}} + j - 1)t/n), \quad S_{tj} = \sin(2\pi(f_{\text{min}} + j - 1)t/n),
\]

respectively.

To ensure that the signs of \( \hat{y} \) are consistent with \( s \), we need the constraints \( s_t a_t^T x \geq 0 \) for \( t = 1, \ldots, n \), where \( a_1^T, \ldots, a_n^T \) are the rows of \( A \). To achieve the proper normalization, we also need the linear equality constraint \( \|\hat{y}\|_1 = s^T Ax = n \). (Note that an \( \ell_1 \)-norm equality constraint is not convex in general, but here it is, since the signs are given.)

We have a convex objective and linear inequality and equality constraints, so our optimization problem is convex:

\[
\begin{align*}
\text{minimize} & \quad \|Ax\|_2 \\
\text{subject to} & \quad s_t a_t^T x \geq 0, \quad t = 1, \ldots, n \\
& \quad s^T Ax = n.
\end{align*}
\]

We get our estimate as \( \hat{y} = Ax^* \), where \( x^* \) is a solution of this problem.

One common mistake was to formulate the problem above without the normalization constraint. The (incorrect) argument was that you’d solve the problem, which is homogeneous, and then scale what you get so its \( \ell_1 \) norm is one. This doesn’t work, since the (unique) solution to the homogeneous problem is \( x = 0 \) (since \( x = 0 \) is feasible). However, this method did give numerical results far better than \( x = 0 \). The reason is that the solvers returned a very small \( x \), for which \( Ax \) had the right sign. And no, that does not mean the error wasn’t a bad one.

ii. The recovery error is 0.1208. This is very impressive considering how little information we were given.

The following matlab code solves the problem:

```matlab
c = zeros(n,B);
s = zeros(n,B);
for j = 1:B
    c(:,j) = cos(2*pi * (f_min+j-1) * (1:n) / n);
    s(:,j) = sin(2*pi * (f_min+j-1) * (1:n) / n);
end
A = [C S];
```

The following matlab code solves the problem:

```matlab
zero_crossings_data;

% Construct matrix A whose columns are bandlimited sinusoids
C = zeros(n,B);
s = zeros(n,B);
for j = 1:B
    C(:,j) = cos(2*pi * (f_min+j-1) * (1:n) / n);
    s(:,j) = sin(2*pi * (f_min+j-1) * (1:n) / n);
end
A = [C S];
```
Figure 1 The original bandlimited signal $y$ and the estimate $\hat{y}$ recovered from zero crossings.

```
% Minimize norm subject to L1 normalization and sign constraints
cvx_begin quiet
    variable x(2*B)
    minimize norm(A*x)
    subject to
        s .* (A*x) >= 0
        s' * (A*x) == n
cvx_end

y_hat = A*x;
fprintf('Recovery error: %f\n', norm(y - y_hat) / norm(y));
figure
plot(y)
hold all
plot(y_hat)
xlim([0,n])
legend('original', 'recovered', 'Location', 'SouthEast');
title('original and recovered bandlimited signals');
```

The following Python code solves the problem:
import numpy as np
import cvxpy as cvx
import matplotlib.pyplot as plt

from zero_crossings_data import *

# Construct matrix A whose columns are bandlimited sinusoids
C = np.zeros((n, B))
S = np.zeros((n, B))
for j in range(B):
    C[:, j] = np.cos(2 * np.pi * (f_min + j) * np.arange(1, n + 1) / n)
    S[:, j] = np.sin(2 * np.pi * (f_min + j) * np.arange(1, n + 1) / n)
A = np.hstack((C, S))

# Minimize norm subject to L1 normalization and sign constraints
x = cvx.Variable(2 * B)
obj = cvx.norm(A * x)
constraints = [cvx.mul_elemwise(s, A * x) >= 0,
               s.T * (A * x) == n]
problem = cvx.Problem(cvx.Minimize(obj), constraints)
problem.solve()
y_hat = np.dot(A, x.value.A1)
print('Recovery error: {}' .format(np.linalg.norm(y - y_hat) / np.linalg.norm(y)))

plt.figure()
plt.plot(np.arange(0, n), y, label='original');
plt.plot(np.arange(0, n), y_hat, label='recovered');
plt.xlim([0, n])
plt.legend(loc='lower left')
plt.show()

The following Julia code solves the problem:
using Convex, SCS, Gadfly
set_default_solver(SCSSolver(verbose=false))

include("zero_crossings_data.jl")

# Construct matrix A whose columns are bandlimited sinusoids
C = zeros(n, B);
S = zeros(n, B);
for j in 1:B
    C[:, j] = cos(2 * pi * (f_min + j - 1) * (1:n) / n);
S[:, j] = sin(2 * pi * (f_min + j - 1) * (1:n) / n);
end
A = [C S];

# Minimize norm subject to L1 normalization and sign constraints
x = Variable(2 * B)
obj = norm(A * x, 2)
constraints = [s .* (A * x) >= 0,
               s' * (A * x) == n]
problem = minimize(obj, constraints)
solve!(problem)

y_hat = A * x.value
println("Recovery error: $(norm(y - y_hat) / norm(y))")
pl = plot(
    layer(x=1:n, y=y, Geom.line, Theme(default_color = colorant"blue")),
    layer(x=1:n, y=y_hat, Geom.line, Theme(default_color = colorant"green")),
);
display(pl);