HW10

May 15, 2020

1 HW10

1.1 alireza darvishi 96109674

1.1.1 3.3:

loading libraries and declaring Variables

[1]:
```
import cvxpy as cp
import numpy as np
import matplotlib.pyplot as plt
x = cp.Variable()
y = cp.Variable()
z = cp.Variable()
```

(a) cp.norm([x + 2 * y , x - y]) == 0
(b) square (square (x + y)) <= x - y
(c) 1/x + 1/y <= 1 ; x >= 0 ; y >= 0
(d) norm([max(x , 1) , max (y , 2)]) <= 3 * x + y
(e) x * y >= 1 ; x >= 0 ; y >= 0
(f) (x + y)^2/sqrt(y) <= x - y + 5
(g) x^3 + y^3 <= 1 ; x >= 0 ; y >= 0
(h) x + z <= 1 + sqrt(x * y - z^2) ; x >= 0 ; y >= 0

a) first we introduce a new variable t with help from np array then we set inequality of norm(t) $\leq$ 0 because setting norm(t) == 0 is not dcp and left hand is not affien but norm (t) $\geq$ holds already

[2]:
```
t=x*[1,1]+y*[2,-1]
constrain  = (cp.norm(x) <= 0)
print(constrain.is_dcp())
prob = cp.Problem(cp.Minimize(x),[constrain])
print(prob.solve())
```

```
True
-4.841433171885098e-10
```

b) replace square of square with power function

[3]:
```
constrain  = (cp.power((x+y),4) <= x-y)
print(constrain.is_dcp())
prob = cp.Problem(cp.Minimize(x),[constrain])
print(prob.solve())
```

```
True
-0.23623519704504156
```

c) declare 2×1 vector t with x and y in each replace 1/x + 1/y with harmonic_mean function

[4]:
```
t=x*[1,0]+y*[0,1]
constraints = [cp.harmonic_mean(t)>=2, x >= 0, y >= 0]
print(constraints[0].is_dcp())
prob = cp.Problem(cp.Minimize(x),constraints)
print(prob.solve())
```

```
True
1.0004149249896446
```

d) t = max (1, x) so 1<=t and x<=t u = max (y, 2) so y<=u and 2<=u finally v=[u,t] and norm(c)<= 3*x+y

[5]:
```
t=cp.Variable()
u=cp.Variable()
v=t*[1,0]+u*[0,1]
constraints = [1<=t , x<=t , y <= u , 2 <= u , cp.norm(v)<=3*x+y]
prob = cp.Problem(cp.Minimize(y),constraints)
print(prob.solve())
```

```
-inf
```

e) x>=1/y

[6]:
```
constraints = [x>=pow(y,-1),x>=0 , y>= 0]
prob = cp.Problem(cp.Minimize(x),constraints)
print(prob.solve())
```

```
0.00033978381252083774
```

f)

[7]:
```
t=(x+y)
u=cp.sqrt(y)
v=cp.quad_over_lin(t,u)
constraints = [v<=x-y+5]
prob = cp.Problem(cp.Minimize(x),constraints)
print(prob.solve())
```

```
-2.6766326052203664
```

g) simply use p-norm with p=3

[8]:
```
t=x*[1,0]+y*[0,1]
constraints = [cp.norm(t,3) <= 1, x >= 0, y >= 0]
prob = cp.Problem(cp.Minimize(x),constraints)
print(prob.solve())
```

```
-9.08091690198461e-12
```

h) t = z^2/y u = $\sqrt{y(x-t)}$ x+z <= 1+u and all expressions are dcp

[9]:
```
t=cp.quad_over_lin(z,y)
u=cp.geo_mean( (x-t)*[1,0]+y*[0,1])
constraints=[x+z <= 1+u ]
prob = cp.Problem(cp.Minimize(x),constraints)
print(prob.solve())
```

```
1.508057932883562e-09
```

1.1.2 3.32 c

[10]:
```
from satisfy_some_constraints_data import *

x = cp.Variable(n)
v = cp.Variable()
objective = cp.Minimize(c.T @ x)
constraints = [ v >=0 , cp.sum(cp.pos( v+ A @ x - b ))<= (m-k)*v ]
prob = cp.Problem( objective ,constraints )
print("optimal value is:",prob.solve())
print("lambda is:",1/v.value)
print("number of satisfied constraints:", sum( A @ x.value - b <= 10**-5))

idx = np.argsort(A @ x.value - b)[:k]
new_const = [ (A @ x - b)[idx[:-2]]<= 0 ]
new_prob = cp.Problem (objective , new_const )
print("new optimal value:",new_prob.solve())
```

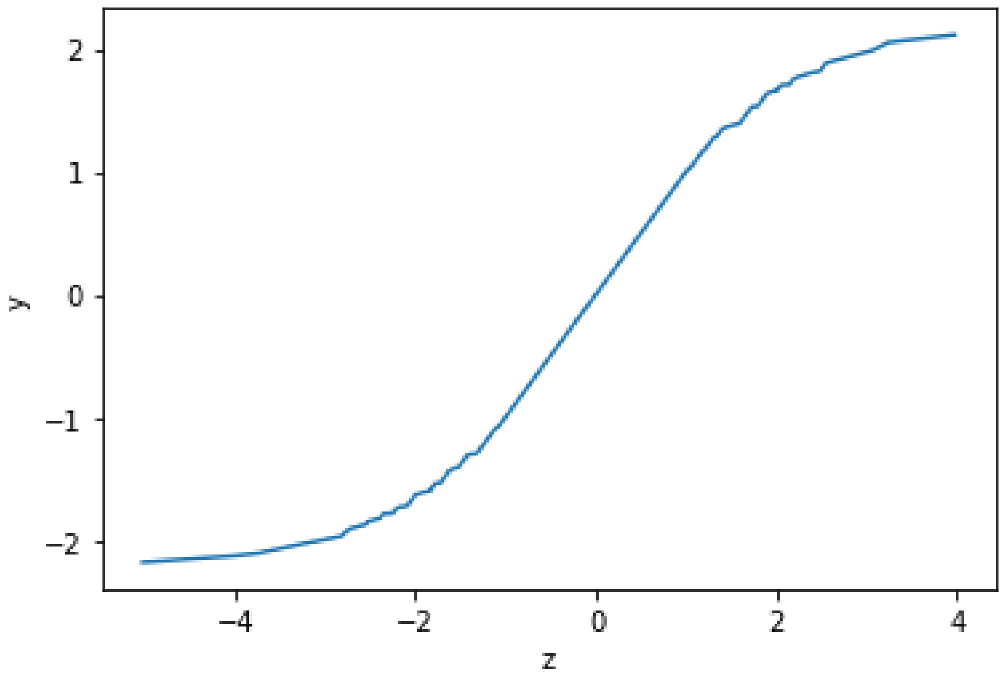
```
optimal value is: -8.454464943573852
lambda is: 282.9846637002597
number of satisfied constraints: 66
new optimal value: -8.891600097480849
```

1.1.3 6.5 b

[11]:
```
from nonlin_meas_data import *
x = cp.Variable(n)
z = cp.Variable(m)
objective = cp.Minimize( cp.norm( z - A @ x ) )
```

```
constraints = [ z[1:]-z[:-1] <= (y[1:]-y[:-1])/alpha , z[1:]-z[:-1] >= (y[1:
↪]-y[:-1])/beta ]
prob = cp.Problem( objective ,constraints )
prob.solve()
print("x estimation is:", x.value)
plt.plot(z.value,y);
plt.ylabel("y");
plt.xlabel("z");
```

```
x estimation is: [ 0.48194427 -0.46569465  0.9364119   0.92966369]
```



علیرضا درفشی - 96109679 - تمرین ۱۰ - بهینه سازی محدب

3.10

الف)

$E\,C^T x = C_0^T x$

$$\Rightarrow \left.\begin{array}{l}\min\ C_0^T x \\ \text{st. } Ax \le b\end{array}\right\} \rightarrow$$ یک مسئله‌ی برنامه‌ریزی خطی

ب)

$E\,C^T x + \gamma \operatorname{Var}(C^T x) = C_0^T x + \gamma\, x^T C' C^T x - \gamma x^T C_0 C_0^T x$

$= C_0^T x + \gamma\, x^T \Sigma x$

$$\Rightarrow \left.\begin{array}{l}\min\ C_0^T x + \gamma\, x^T \Sigma x \\ \text{st: } Ax \le b\end{array}\right\} \rightarrow$$ یک مسئله‌ی QP است.

برای $\gamma \ge 0$

ج) خیر. به ازای $\gamma < 0$، تابع هدف concave است.

د) $C^T x$ یک متغیر نرمال است که: $C^T x \sim N(C_0^T x, x^T \Sigma x)$

$$\Rightarrow \frac{C^T x - C_0^T x}{\|\Sigma^{\frac{1}{2}} x\|_2} \sim N(0,1) \Rightarrow P(C^T x > B) \le \alpha : P\left(\frac{C^T x - C^T x_0}{\|\Sigma^{\frac{1}{2}} x\|_2} > \frac{B - C_0^T x}{\|\Sigma^{\frac{1}{2}} x\|_2}\right) \le \alpha$$

$$\Rightarrow 1 - \phi\left(\frac{B - C_0^T x}{\|\Sigma^{\frac{1}{2}} x\|_2}\right) \le \alpha \Rightarrow (1-\alpha) \le \phi\left(\frac{B - C_0^T x}{\|\Sigma^{\frac{1}{2}} x\|_2}\right)$$

$$\Rightarrow \Phi^{-1}(1-\alpha) \leq \frac{\beta - C_0^T x}{\|\Sigma^{\frac{1}{2}} x\|_2}$$

$$\Rightarrow \|\Sigma^{\frac{1}{2}} x\|_2 \, \Phi^{-1}(1-\alpha) + C_0^T x \leq \beta$$

$$\Rightarrow \quad \min \ \beta$$

$$\text{st.} \quad \Phi^{-1}(1-\alpha) \, \|\Sigma^{\frac{1}{2}} x\|_2 + C_0^T x \leq \beta$$

$$Ax \leq b$$

که یک مسئله‌ی SOCP است اگر $\alpha \leq 0.5$ درنتیجه $\Phi^{-1}(1-\alpha) \geq 0$

اگر $\alpha \geq 0.5$ مسئله‌ی خنثای ریسک را حل می‌کنیم

3.32

الف)

$$(1+\lambda u)_+ \geq 1(u>0) = 1 - 1(u\leq 0)$$

$$\Rightarrow \sum_{i=1}^{m}(1+\lambda f_i(x))_+ \geq m - \underbrace{\sum 1(f_i(x)\leq 0)}_{\text{تعداد مسائل های که برقرار هستند}}$$

اگر شرط گفته شده برقرار باشد:

$$\Rightarrow \sum_{i=1}^{m}(1+\lambda f_i(x))_+ \leq m-k$$

$$\Rightarrow m - \sum 1(f_i(x)\leq 0)_+ \leq m-k \Rightarrow k \leq 1(f_i(x)\leq 0)$$

تعداد شرط های برقرار شده از k بزرگتر است ✓

ب)

$$\lambda>0 \Rightarrow (1+\lambda f_i(x))_+ = \lambda\left(\frac{1}{\lambda}+f_i(x)\right)_+$$

$$\Rightarrow \sum_{i=1}^{m}(1+\lambda f_i(x))_+ \leq m-k \Rightarrow \lambda\sum_{i=1}^{m}\left(\frac{1}{\lambda}+f_i(x)\right)_+ \leq m-k$$

$$\Rightarrow \sum_{i=1}^{m}\left(\frac{1}{\lambda}+f_i(x)\right)_+ \leq (m-k)\frac{1}{\lambda}, \quad \frac{1}{\lambda}=v$$

$$\Rightarrow \sum_{i=1}^{m}(v+f_i(x))_+ \leq v(m-k)$$

$\Rightarrow \min \quad f_0(x)$

$\text{st.} \quad \sum_{i=1}^{m} (v + f_i(x))_+ \le v(m-k)$

$v \ge 0$

دلیل اضافه کردن $v=0$ هم این است که با $v=0$ ، همه‌ی شرط‌ها برقرار هستند.

$\min \quad \sum_k x_k \log\left(\frac{x_k}{y_k}\right)$ 4.3

$\text{st} \quad Ax = b \,, \quad 1^T x = 1$

$$g(x, v, w) = \sum_k x_k \log\left(\frac{x_k}{y_k}\right) - v^T(Ax-b) - w(1^T x - 1)$$

$$\frac{\partial g}{\partial x_k} = \log\left(\frac{x_k}{y_k}\right) + 1 - (A^T v)_k - w = \log\left(\frac{x_k}{y_k}\right) + 1 - a_k^T v - w = 0$$

$$\Rightarrow x_k = y_k e^{(w + a_k^T v - 1)}$$

$$\Rightarrow g = \sum_k y_k e^{(w + a_k^T v - 1)} (w + a_k^T v - 1) - \sum (A^T v)_k x_k + v^T b$$

$$- w \sum x_k + w$$

$$= \sum_k y_k e^{(w + a_k^T v - 1)} (w + a_k^T v - 1) - \sum (a_k^T v) y_k e^{(w + a_k^T v - 1)}$$

$$+ v^T b - w \sum y_k e^{(w + a_k^T v - 1)} + w$$

$$\Rightarrow g(\nu, w) = \nu^T b + w - \sum_k y_k e^{(w + a_k^T \nu - 1)}$$

$$\frac{\partial g}{\partial w} = 1 - \sum_k y_k e^{(w + a_k^T \nu - 1)} = 1 - e^w \sum_k y_k e^{(a_k^T \nu - 1)}$$

$$\Rightarrow e^w = \frac{1}{\sum_k y_k e^{(a_k^T \nu - 1)}}, \quad w = -\log\Big(\sum_k y_k e^{(a_k^T \nu - 1)}\Big)$$

$$\Rightarrow g(\nu) = \nu^T b - \log\Big(\sum_k y_k e^{(a_k^T \nu - 1)}\Big) - \frac{\sum_k y_k e^{(a_k^T \nu - 1)}}{\sum_k y_k e^{(a_k^T \nu - 1)}}$$

$$= \nu^T b - \log\Big(\sum_k y_k e^{a_k^T \nu}\Big) + 1 - 1$$

$\Rightarrow$ dual problem: $\max \quad \nu^T b - \log\Big(\sum_k y_k e^{a_k^T \nu}\Big)$

ok.

6.5

$$z_i = a_i^T x + v_i \Rightarrow v_i = z_i - a_i^T x$$

الف)

$$f_0 = \sum_i \log\left(P(v = v_i)\right)$$

$$= \text{Const} - \frac{1}{2\sigma^2} \sum_i (z_i - a_i^T x)^2$$

$$z_{i+1} - z_i \le \frac{\beta(y_{i+1} - y_i)}{\alpha} \quad \text{for } i = 1 \text{ to } N-1$$

$$z_{i+1} - z_i \ge \frac{\alpha(y_{i+1} - y_i)}{\beta} \quad \text{for } i = 1 \text{ to } N-1$$

$$\Rightarrow \min \sum_i (z_i - a_i^T x)^2$$

st:

$$z_{i+1} - z_i \le \frac{\beta(y_{i+1} - y_i)}{\alpha} \quad \text{for } i = 1 \text{ to } N-1$$

$$z_{i+1} - z_i \ge \frac{\alpha(y_{i+1} - y_i)}{\beta} \quad \text{for } i = 1 \text{ to } N-1$$