CMSE890 Homework#3

Haiyang Yu

October 30, 2019

1.

(1)

$f(\boldsymbol{x}) = ||\boldsymbol{x}||_1$, so

$$f^*(\boldsymbol{y}) = \begin{cases} +\infty, & ||\boldsymbol{y}||_\infty > 1 \\ 0, & ||\boldsymbol{y}||_\infty \leq 1 \end{cases}$$

Define

$$h(\boldsymbol{y}) = f^*(\boldsymbol{y}) + d(\boldsymbol{y}) = \begin{cases} \mu \sum_{i=1}^{n} (1 - \sqrt{1 - y_i^2}), & ||\boldsymbol{y}||_\infty \leq 1 \\ +\infty, & ||\boldsymbol{y}||_\infty > 1 \end{cases}$$

Thus,

$$\begin{aligned} h^*(\boldsymbol{x}) &= \sup_{\boldsymbol{y}} \boldsymbol{x}^{\mathrm{T}} \boldsymbol{y} - h(\boldsymbol{y}) \\ &= \sup_{||\boldsymbol{y}||_\infty \leq 1} \boldsymbol{x}^{\mathrm{T}} \boldsymbol{y} - \mu \sum_{i=1}^{n} (1 - \sqrt{1 - y_i^2}) \\ &= \sum_{i=1}^{n} \sup_{|y_i| \leq 1} \left\{ x_i y_i - \mu (1 - \sqrt{1 - y_i^2}) \right\} \\ &= \sum_{i=1}^{n} \left[x_i y_i - \mu (1 - \sqrt{1 - y_i^2}) \right]_{y_i = x_i / \sqrt{\mu^2 + x_i^2}} \\ &= \sum_{i=1}^{n} \sqrt{x_i^2 + \mu^2} - \mu \end{aligned}$$

(2)

$f(\boldsymbol{x}) = \max_i x_i$, so

$$f^*(\boldsymbol{y}) = \begin{cases} 0, & \text{if } \sum_{i=1}^{n} y_i = 1, \text{ and } y_i \geq 0 \text{ for all } i \\ +\infty, & \text{else} \end{cases}$$

Define

$$h(\boldsymbol{y}) = f^*(\boldsymbol{y}) + d(\boldsymbol{y}) = \begin{cases} \mu \sum_{i=1}^{n} (y_i \log y_i + \log n), & \text{if } \sum_{i=1}^{n} y_i = 1, \text{ and } y_i \geq 0 \text{ for all } i \\ +\infty, & \text{else} \end{cases}$$

Thus,

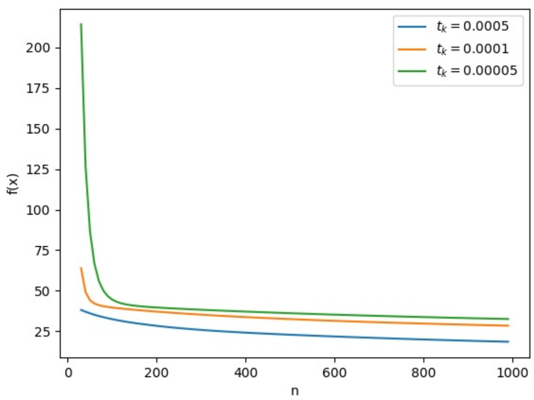
$$\begin{aligned}
h^*(\boldsymbol{x}) &= \sup_{\boldsymbol{y}} \boldsymbol{x}^{\mathrm{T}}\boldsymbol{y} - h(\boldsymbol{y}) \\
&= \sup_{\substack{\sum_{i=1}^n y_i=1 \\ y_i \geq 0}} \left\{ \boldsymbol{x}^{\mathrm{T}}\boldsymbol{y} - \mu \sum_{i=1}^n (y_i \log y_i + \log n) \right\} \\
&= \sum_{i=1}^n x_i y_i - \mu y_i \log y_i - \mu \log n \Big|_{y_i = \frac{\mathrm{e}^{\frac{x_i}{\mu}-1}}{\sum_{i=1}^n \mathrm{e}^{\frac{x_i}{\mu}-1}}} \\
&= \mu + \mu \log \left(\sum_{i=1}^n \mathrm{e}^{\frac{x_i}{\mu}-1} \right) - \mu n \log n
\end{aligned}$$

2.

$f(X) = -\log\det X$.Suppose $X = P^{\mathrm{T}}\Lambda P$, where P is orthogonal matrix and $\Lambda = \mathrm{diag}(\lambda_1, \lambda_2, \cdots, \lambda_n)$ is a diagonal matrix with positive eigenvalues. U has positive eigenvalues u_i.

$$\begin{aligned}
\mathrm{prox}_f(X) &= \arg\min_U -\log\det U + \frac{1}{2}||U - X||_F^2 \\
&= \arg\min_U -\log\det PUP^{-1} + \frac{1}{2}||P^{-1}(PUP^{-1} - \Lambda)P||_F^2 \\
&= P^{-1}\left(\arg\min_U -\log\det U + \frac{1}{2}||U - \Lambda||_F^2 \right) P \\
&= P^{-1}\left(\arg\min_U \sum_{i=1}^n -\log u_i + \frac{1}{2}(u_i - \lambda_i)^2 \right) P \\
&= P^{-1}\mathrm{diag}(\frac{\lambda_1 + \sqrt{\lambda_1^2 + 4}}{2}, \frac{\lambda_2 + \sqrt{\lambda_2^2 + 4}}{2}, \cdots, \frac{\lambda_{\mathrm{n}} + \sqrt{\lambda_{\mathrm{n}}^2 + 4}}{2})P
\end{aligned}$$

(3)




```
import numpy as np
import matplotlib.pyplot as plt

def partial(x):
    tmp=A.dot(x)-b
    res=A.T.dot(tmp)
```

```
    return res

def prox(t,x):    #when h(x)=||x||_{1}
    res=np.zeros((len(x),1))
    for i in range(len(x)):
        if x[i]>t:
            res[i]=x[i]-t
        if x[i]<-t:
            res[i]=x[i]+t
    return res

def f(x):
    s1=0
    for i in range(len(x)):
        s1+=abs(x[i])
    tmp=A.dot(x)-b
    s2=np.dot(tmp.T,tmp)
    return s1+s2


np.random.seed(1000)
m=200
n=1000
A=np.random.normal(0,1,(m,n))
x_bar=np.zeros((n,1))
for i in range(20):
    k=np.random.randint(0,1000)
    x_bar[k]=np.random.normal(0,1)
b=A.dot(x_bar)

x_pre=np.zeros((n,1))
y1=[]
n1=[]
y2=[]
n2=[]
y3=[]
n3=[]
iteration=1000
for i in range(iteration):
    tk=0.0005
    x_now=prox(tk,x_pre-tk*partial(x_pre))
    x_pre=x_now
    if i%10==0 and i>20:
        y1.append(f(x_now)[0][0])
        n1.append(i)
x_pre=np.zeros((n,1))
for i in range(iteration):
    tk=0.0001
    x_now=prox(tk,x_pre-tk*partial(x_pre))
    x_pre=x_now
    if i%10==0 and i>20:
        y2.append(f(x_now)[0][0])
        n2.append(i)
x_pre=np.zeros((n,1))
for i in range(iteration):
    tk=0.00005
```

```
    x_now=prox(tk,x_pre-tk*partial(x_pre))
    x_pre=x_now
    if i%10==0 and i>20:
        y3.append(f(x_now)[0][0])
        n3.append(i)
plt.plot(n1,y1,label=r'$t_{k}=0.0005$')
plt.plot(n2,y2,label=r'$t_{k}=0.0001$')
plt.plot(n3,y3,label=r'$t_{k}=0.00005$')
plt.xlabel("n")
plt.ylabel("f(x)")
plt.legend()
plt.show()
```