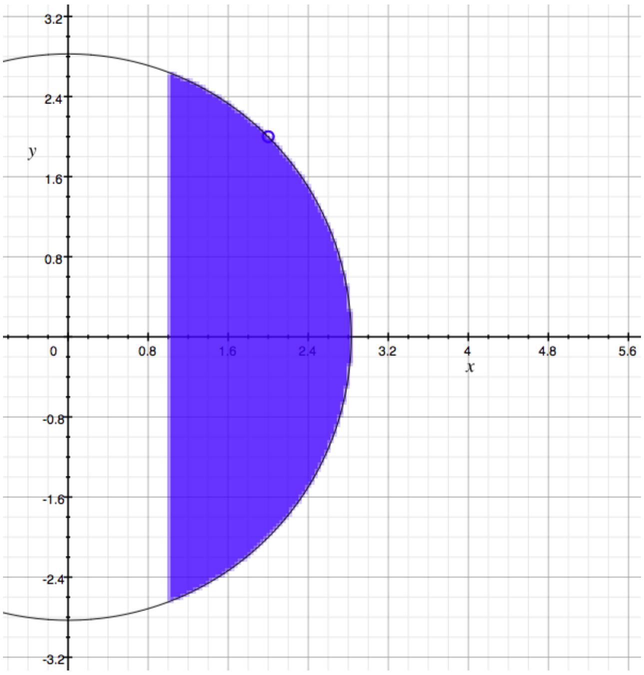
Exercise 9.3 [Boyd & Vandenberghe, 2004]

Ans: (a)

$$p^* = \inf_{x_1 > 1, x_2 \in \mathbb{R}} x_1^2 + x_2^2 = 1$$

Yet the minimum cannot be attained since we require $x_1 > 1$.

(b) Mac built-in Grapher is employed to generate the following graph for the sub-level set $S = \{x \mid f(x) \leq f(x^{(0)})\}$.



We can see that this is the intersection of the open half-plane $\{x \mid x_1 > 1\}$ with a closed disk $\{x \mid \|x\| \leq \sqrt{8}\}$ which is not closed. On the other hand, $\nabla^2 f = I$ is strongly convex regardless of the domain.

(c) We have $\nabla f(x) = 2x$; starting off at $x^{(0)} = (2, 2)$, the gradient will always bring us towards the origin along the line $x_1 = x_2$ since $x^{(k+1)} = x^{(k)} - \alpha_k \nabla f(x^{(k)}) = (1 - 2\alpha_k)x^{(k)}$. This shows us that the gradient decent method, although implemented with backtracking line search, can only approach a minimum value of $f(1, 1) = 2$, which is significantly larger than the actual optimal value $p^* = 1$. □

Additional Exercise 8.1 [Boyd & Vandenberghe, 2017]

Ans: (a) First from Cauchy's inequality,

$$x_1y_1 + \sqrt{\gamma}x_2y_2 \leq \sqrt{x_1^2+\gamma x_2^2}\sqrt{y_1^2+y_2^2} \leq \sqrt{x_1^2+\gamma x_2^2}$$

This upper bound can be attained when $y_1 = \lambda x_1, y_2 = \lambda\sqrt{\gamma}x_2$. From the constraint $y_1^2 + y_2^2 \leq 1$, we get

$$\lambda^2(x_1^2+\gamma x_2^2) \leq 1$$

or $\lambda \leq \frac{1}{\sqrt{x_1^2+\gamma x_2^2}}$. Substituting this result to the constraint $y_1 = \lambda x_1 \geq \frac{1}{\sqrt{1+\gamma}}$, we get

$$\begin{aligned} \frac{x_1}{\sqrt{x_1^2+\gamma x_2^2}} &\geq \frac{1}{\sqrt{1+\gamma}} \\ (1+\gamma)x_1^2 &\geq x_1^2+\gamma x_2^2 \\ x_1^2 &\geq x_2^2 \end{aligned}$$

Along with the constraint $y_1 \geq \frac{1}{\sqrt{1+\gamma}} > 0$ from which we can induced $x_1 \geq 0$ (o.w. $x_1y_1 \leq 0$ can't make it to the maximum!), we conclude that maximum $\sqrt{x_1^2+\gamma x_2}$ is attained when

$$x_1 \geq |x_2|$$

When this constraint is not satisfied, we can see that $\sqrt{\gamma}x_2$ has a larger leverage than x_1 in the expression $x_1y_1 + \sqrt{\gamma}x_2y_2$ when we have to balance the weights y_1, y_2 due to the constraint $y_1^2 + y_2^2 \leq 1$; therefore to maximize this term, we choose the smallest possible $y_1 = \frac{1}{\sqrt{1+\gamma}}$, then the term is maximized when $y_2 = sgn(x_2)\frac{\sqrt{\gamma}}{\sqrt{1+\gamma}}$ with a value

$$\frac{x_1}{\sqrt{1+\gamma}} + \frac{|x_2|\sqrt{\gamma}}{\sqrt{1+\gamma}} = \frac{x_1+|x_2|\sqrt{\gamma}}{\sqrt{1+\gamma}}$$

as desired. Now that $f(x_1, x_2)$ is the supremum of an affine expression over a convex set $D = \{y_1^2 + y_2^2 \leq 1\} \cap \{y_1 \geq 1/\sqrt{1+\gamma}\}$, f itself is a convex function. Note that f is unbounded below since when $x_2 = 0, 0 > x_1 \to -\infty$,

$$f(x_1, x_2) \leq x_2\sqrt{1+\gamma} \to \infty$$

(b) First we calculate the derivatives of f,

$$\frac{\partial f}{\partial x_1} = \begin{cases} \dfrac{x_1}{\sqrt{x_1^2+\gamma x_2^2}}, & |x_2| \leq x_1 \\ \dfrac{1}{\sqrt{1+\gamma}}, & \text{otherwise} \end{cases}, \frac{\partial f}{\partial x_2} = \begin{cases} \dfrac{\gamma x_2}{\sqrt{x_1^2+\gamma x_2^2}}, & |x_2| \leq x_1 \\ \dfrac{\gamma sgn(x_2)}{\sqrt{1+\gamma}}, & \text{otherwise} \end{cases}$$

When we start at $x^{(0)} = (\gamma, 1)$, surely

$$x_1^{(0)} = \gamma = \gamma\left(\frac{\gamma-1}{\gamma+1}\right)^0, x_2^{(0)} = 1 = \left(-\frac{\gamma-1}{\gamma+1}\right)^0$$

This forms the base step of the mathematical induction. Now assuming $x_1^{(k)} = \gamma\left(\frac{\gamma-1}{\gamma+1}\right)^k$, $x_2^{(k)} = \left(-\frac{\gamma-1}{\gamma+1}\right)^k = \frac{1}{\gamma}(-1)^k x_1^{(k)}$, then we fall into the case that $|x_2^{(k)}| \le x_1^{(k)}$ and

$$\nabla f(x^{(k)}) = \left(\frac{1}{\sqrt{1+\frac{1}{\gamma}}}, \frac{(-1)^k}{\sqrt{1+\frac{1}{\gamma}}}\right) = \left(\frac{\sqrt{\gamma}}{\sqrt{\gamma+1}}, \frac{(-1)^k\sqrt{\gamma}}{\sqrt{\gamma+\gamma}}\right)$$

Here to simply the notation, we denote $L = \frac{\gamma-1}{\gamma+1}$, $H = \sqrt{\frac{\gamma}{\gamma+1}}$, then $x^{(k)} = (\gamma, (-1)^k)L^k$, $\nabla f(x^{(k)}) = H(1, (-1)^k)$. Now to perform the exact line search, we take the infimum over all possible $t \ge 0$ and complete the square,

$$\begin{aligned}
f(x^{(k)} - t\nabla f(x^{(k)})) &= f(\gamma L^k - tH, (-1)^k L^k - (-1)^k tH) \\
&= \sqrt{(\gamma L^k - tH)^2 + \gamma(L^k - tH)^2} \\
&= \sqrt{\gamma^2 L^{2k} - 2\gamma t L^k H + t^2 H^2 + \gamma L^{2k} - 2\gamma t L^k H + \gamma t^2 H^2} \\
&= \sqrt{(1+\gamma)H^2 t^2 - 4\gamma t L^k H + (\gamma^2+\gamma)L^{2k}} \\
&= \sqrt{(1+\gamma)H^2\left(t - \frac{4\gamma L^k H}{2(1+\gamma)H^2}\right)^2 + (\text{some unimportant constant})}
\end{aligned}$$

Hence we get the optimal $t^* = \frac{2\gamma L^k}{(1+\gamma)H}$ (nonnegative indeed!) and

$$\begin{aligned}
x^{(k+1)} &= x^{(k)} - \frac{2\gamma L^k}{(1+\gamma)H}\nabla f(x^{(k)}) \\
&= \left(\gamma L^k - \frac{2\gamma L^k}{1+\gamma}, (-1)^k L^k - (-1)^k \frac{2\gamma L^k}{1+\gamma}\right) \\
&= \left(\gamma L^k\left(1 - \frac{2}{1+\gamma}\right), (-1)^k L^k\left(1 - \frac{2\gamma}{1+\gamma}\right)\right) \\
&= \left(\gamma L^k\left(\frac{\gamma-1}{1+\gamma}\right), (-1)^k L^k\left(\frac{1-\gamma}{1+\gamma}\right)\right) \\
&= (\gamma L^{k+1}, (-1)^{k+1} L^{k+1})
\end{aligned}$$

as desired. Now that $L = \frac{\gamma-1}{\gamma+1} < 1$, we know $x^{(k)} = (\gamma, (-1)^k)L^k \to 0$ as $k \to \infty$. Certainly gradient decent method with exact line search doesn't give correct optimum for this problem. □

Additional Exercise 8.9 [Boyd & Vandenberghe, 2017]

Ans: (a) Define

$$Q(s) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{s} e^{-\frac{t^2}{2}} dt$$

In homework 3, **Exercise 3.55 [Boyd & Vandenberghe]**, we proved this would a log-concave function since $h(t) = \frac{t^2}{2}$ is convex. Now we can write the objective function in terms of Q,

$$l(x) = \sum_{b_i - a_i^T x \le 0} \log(1 - Q(b_i - a_i^T x)) + \sum_{b_i - a_i^T x > 0} \log Q(b_i - a_i^T x)$$

Another observation is for $1 - Q(s) = Q(-s)$ since $e^{-\frac{t^2}{2}}$ is an even function; if we let $\tilde{a}_i^T = sgn(b_i - a_i^T x) a_i^T$, $\tilde{b}_i = sgn(b_i - a_i^T x) \tilde{b}_i$, then objective function can be further rewritten

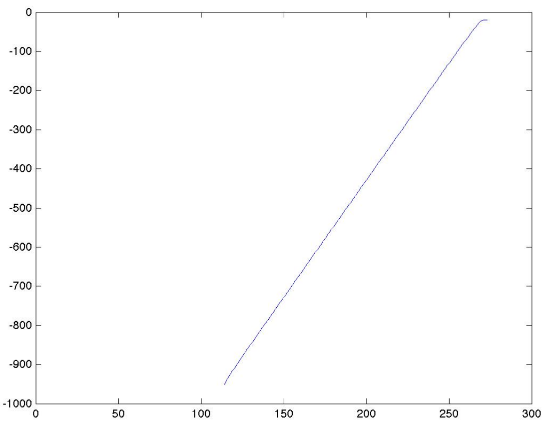
$$l(x) = \sum_{i=1}^{m} \log Q(\tilde{b}_i - \tilde{a}_i^T x)$$

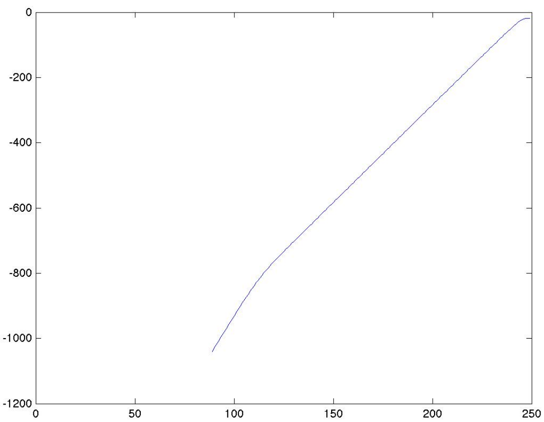
as a sum of concave functions composite affine functions on x, which is concave.

(b) To apply Newton's method on this problem, we first calculate the gradient and Hessian of the objective function $f(x) = -l(x)$. Denote $z = \tilde{b} - \tilde{A}x$ for simplicity.

$$\begin{aligned}
\nabla f(x) &= -\sum_{i=1}^{m} \frac{Q'(\tilde{b}_i - \tilde{a}_i^T x)}{Q(\tilde{b}_i - \tilde{a}_i^T x)} (-\tilde{a}_i) \\
&= \sum_{i=1}^{m} \frac{e^{-z_i^2/2}}{\int_{-\infty}^{z_i} e^{-t^2/2} dt} \tilde{a}_i \\
&= \sum_{i=1}^{m} \frac{1}{\sqrt{2\pi} \cdot \frac{1}{2} \mathbf{erfcx}(-z_i/\sqrt{2})} \tilde{a}_i \\
\nabla^2 f(x) &= -\sum_{i=1}^{m} \frac{Q''Q - (Q')^2}{Q^2} \tilde{a}\tilde{a}^T \\
&= \sum_{i=1}^{m} \frac{s e^{-\frac{s^2}{2}} \int_{-\infty}^{s} e^{-\frac{t^2}{2}} dt + e^{-s^2}}{\left(\int_{-\infty}^{s} e^{-t^2/2} dt \right)^2} \tilde{a}_i \tilde{a}_i^T \\
&= \sum_{i=1}^{m} \left(\frac{s e^{-s^2/2}}{\int_{-\infty}^{s} e^{-t^2/2} dt} + \left(\frac{e^{-s^2/2}}{\int_{-\infty}^{s} e^{-t^2/2} dt} \right)^2 \right) \tilde{a}_i \tilde{a}_i^T \\
&= \sum_{i=1}^{m} \left(\frac{s}{\sqrt{2\pi} \cdot \frac{1}{2} \mathbf{erfcx}(-s/\sqrt{2})} + \left(\frac{1}{\sqrt{2\pi} \cdot \frac{1}{2} \mathbf{erfcx}(-s/\sqrt{2})} \right)^2 \right) \tilde{a}_i \tilde{a}_i^T
\end{aligned}$$

where $s = \tilde{b}_i - \tilde{a}_i^T x$ in each summand. (Note that from our treatment on coefficients, $\tilde{b}_i - \tilde{a}_i^T x$ is always nonnegative, hence the above Hessian is negative semi-definite, indeed!) The following are two graphs picturing the process of maximizing $l(x)$. Note that I chose the initial guess by random so the graphs look different. I also picked $\alpha = 0.2, \beta = 0.5$. The algorithm usually terminates within about 250 iterations.
Note: I failed many times implementing this algorithm due to the rounding errors introduced by the exponentials. I appreciate how this problem practices me with the MATLAB function `erfcx`.





The optimal solution is around $x = (-0.2708,\ 9.1485,\ 7.9773,\ 6.7027,\ 6.0250,\ 5.0114,\ 4.2996, 2.6767, 2.0187, 0.6842)$. The following is my MATLAB code.

```
one_bit_meas_data;
[m,n] = size(A);
for i=1:50
    if y(i) > 0
        A(i,:) = -A(i,:);
        b(i) = -b(i);
```

```
        y(i) = -y(i);
    end
end
Q = @(s) 0.5*erfc(-s/sqrt(2));
% dQ = @(s) 1/sqrt(2*pi)*exp(-0.5*s^2);
% ddQ = @(s) -s/sqrt(2*pi)*exp(-0.5*s^2);
l = @(x) sum(log(Q(b-A*x)));
f = @(x) -sum(log(Q(b-A*x)));

max_iter = 1000;
TOL = 1e-3;
alpha = 0.2; %(0.01, 0.3)
beta = 0.5; %(0.1, 0.8)
x = rand(n,1);
for iter = 1:max_iter
    z = b-A*x;
    df = zeros(n,1);
    H = zeros(n);
    for i=1:m
        df = df + sqrt(2/pi)/erfcx(-z(i)/sqrt(2)) * A(i,:)';
        H = H + (z(i)*sqrt(2/pi)/erfcx(-z(i)/sqrt(2)) + (sqrt(2/pi)/erfcx(-z(i)/sqrt(
    end
    dx = - H \ df;
    lsq = df' * (-dx);
    if lsq <= 2*TOL
        break
    end
    t = 1;
    while f(x + t*dx) > f(x) - alpha * t * lsq
        t = beta * t;
    end
    x = x + t * dx;
    x_data(:,iter) = x;
    l_data(iter) = l(x);
end
h1 = plot(1:iter-1,l_data);
saveas(h1,'hw9P89','jpg');
```