AMSC/CMSC 660 Quiz 8 , Fall 2003

1. (10) Consider the following code for computing a quasi-Newton direction for minimizing a function whose gradient is $g(x)$.

```
function [p,B] = bfgs(s,y,B,g)
% Given a previous Hessian approximation  B,
% update it using s (the change in x)
% and y (the change in g)
% where g is the gradient at the most recent point.
% and then compute a Newton-like direction p.

B = B - (B*s)*(B*s)'/(s'*B*s) + (y*y')/(y'*s);
p = -B \ g;
```

I believe the code is correct, but it is inefficient. Identify two sources of the inefficiency and propose remedies.

Answer:

- The first line does $2n^2$ divisions. It would be better to add parentheses to drop this to n: `B = B - (B*s)*((B*s)'/(s'*B*s)) + y*(y'/(y'*s))`.
- In the second line, it is silly to refactor the matrix `B` each time, when it is just a rank-1 update of the previous matrix. Instead, update a factorization or (less desirable) update the inverse.
- Forming Bs three times is inefficient. Save Bs as a temporary vector.

2a (5) If we are trying to minimize a function $f(x)$ $(x \in \mathcal{R}^1)$ subject to the constraint $x \geq 0$, we can instead solve the **unconstrained** problem

$$\min_y f(y^2)$$

where $y \in \mathcal{R}^1$. Give one advantage and one disadvantage of this approach.

Answer: Advantage: can use our software for unconstrained optimization. Disadvantage:

- The function becomes more complicated. For instance, if it was originally quadratic, it now becomes quartic.
- Function evaluation is more expensive.
- For every minimizer $\hat{x}$ for the original problem, we now have two minimizers: $\pm\sqrt{\hat{x}}$.

2b. (5) When we compute a search direction for minimizing a function $f(x)$ through the formula $Hp = -g$, where g is the gradient of f evaluated at the current point, why do we want the matrix H to be positive definite?

Answer: Since $p^T g = -p^T H p$, we are assured that $p^T g < 0$ if H is positive definite, and this means we are walking downhill.