AMATH 515: Homework 2
Sid Meka

1. Change of variables, preconditioning:
Given a twice differentiable function $f(x) : \mathbb{R}^n \to \mathbb{R}$, and an invertible matrix $P \in \mathbb{R}^{n \times n}$, consider the change of variables $u = Px$, along with the transformed objective $g(u) = f(P^{-1}u)$.
Since P is invertible, there is a 1 to 1 correspondence between x and u meaning $u = Px$ and $x = P^{-1}u$.

   (a) Compute the gradient $\nabla g(u)$ and the Hessian $\nabla^2 g(u)$.
   Given the transformation $u = Px$ and $g(u) = f(P^{-1}u)$, we compute the gradient and Hessian of $g(u)$:
   Now for $\nabla g(u)$:

      i. Substitute $x = P^{-1}u$ into $f(x)$. That gives us $g(u) = f(P^{-1}u)$.

      ii. Use the fact that
      $$\nabla g(u) = \frac{\partial f}{\partial x} \cdot \frac{\partial x}{\partial u}$$

      $$x = P^{-1}u \qquad \text{(Given)}$$
      $$\frac{\partial x}{\partial u} = P^{-1} \qquad \text{(Differentiating with respect to } u\text{)}$$

      Also, $\frac{\partial f}{\partial x} = \nabla f(x)$ when we evaluate at $x = P^{-1}u$.

      $$\nabla g(u) = \nabla f(P^{-1}u)P^{-1}$$

   Now for $\nabla^2 g(u)$:

      i. Use the chain rule for the second derivative:

      $$\nabla^2 g(u) = \frac{\partial}{\partial u}\left(\nabla g(u)\right)$$

      Substituting $\nabla g(u) = \nabla f(P^{-1}u)P^{-1}$, this becomes:

      $$\nabla^2 g(u) = \frac{\partial}{\partial u}\left(\nabla f(P^{-1}u)\right)P^{-1}$$

      ii. The derivative of $\nabla f(P^{-1}u)$ with respect to u is:

      $$\frac{\partial}{\partial u}\left(\nabla f(P^{-1}u)\right) = \nabla^2 f(P^{-1}u)\frac{\partial (P^{-1}u)}{\partial u}$$

      iii. Since $\frac{\partial (P^{-1}u)}{\partial u} = P^{-1}$, this becomes:

      $$\nabla^2 g(u) = \nabla^2 f(P^{-1}u)P^{-1}P^{-1}$$

      iv. Therefore, the Hessian of $g(u)$ is:

      $$\nabla^2 g(u) = P^{-T}\nabla^2 f(P^{-1}u)P^{-1}$$

(b) Assume in this part that f is strictly convex, which implies that $\nabla^2 f$ is always invertible. Consider applying Newton's method to $f(x)$ starting with x_0, which gives iterates

$$x_{k+1} = x_k - \nabla^2 f(x_k)^{-1} \nabla f(x_k)$$

as well as applying Newton's method to $g(u)$ starting with $u_0 = Px_0$,

$$u_{k+1} = u_k - \nabla^2 g(u_k)^{-1} \nabla g(u_k)$$

Show that

$$Px_k = u_k$$

holds for $k = 1, 2, ...$ This is an important property of Newton's method.
Here is Newton's Update for $f(x)$:

$$x_{k+1} = x_k - \nabla^2 f(x_k)^{-1} \nabla f(x_k)$$

Here is Newton's Update for $g(u)$:

$$u_{k+1} = u_k - \nabla^2 g(u_k)^{-1} \nabla g(u_k)$$

We will continue by Doing a Proof by Induction:

Proof.

Base Case: In our base case, we will start with $k = 0$. Note that we are given $u_0 = Px_0$. Thus, our base case works.
Inductive Step: Assume $Px_k = u_k$. Showing that $Px_{k+1} = u_{k+1}$:
Substituting $x_k = P^{-1}u_k$, we have:

$$x_{k+1} = x_k - \nabla^2 f(x_k)^{-1} \nabla f(x_k)$$

We can now multiply both sides by P:

$$Px_{k+1} = Px_k - P\nabla^2 f(x_k)^{-1} \nabla f(x_k)$$

Substituting $\nabla f(x_k) = P^{-T} \nabla g(u_k)$ and $\nabla^2 f(x_k) = P^{-T} \nabla^2 g(u_k) P^{-1}$, we can express the Newton Step as:

$$Px_{k+1} = u_k - \nabla^2 g(u_k)^{-1} \nabla g(u_k)$$

Note that this matches the updated for u_{k+1}. Thus:

$$Px_{k+1} = u_{k+1}$$

□

(c) Compare gradient descent applied to $f(x)$ with gradient descent applied to $g(u)$ starting with the same initial conditions (e.g. $u_0 = Px_0$).

$$x_{k+1} = x_k - \alpha_k \nabla f(x_k)$$
$$u_{k+1} = u_k - \alpha_k \nabla g(u_k), \quad u_0 = Px_0$$

Show that unlike the situation with Newton's method, the iterates are no longer equivalent.

Remember that $\nabla g(u) = \nabla f(P^{-1}u)P^{-1}$. Therefore, $u_{k+1} = u_k - \alpha_k \nabla f(P^{-1}u_k)P^{-1}$. Using the fact that $x_k = P^{-1}u_k$, we have: $Px_{k+1} = Px_k - \alpha_k P \nabla f(x_k)$. Unlike Newton's method, where the Hessian transformation cancels out the P, in gradient descent, the step size and direction depend on P. This means $Px_k \neq u_k$ for $k \geq 1$.

(d) Now consider applying gradient descent to $g(u)$, starting with u_0 meaning that

$$u_{k+1} = u_k - \alpha_k \nabla g(u_k)$$

Let $\hat{x_k} = P^{-1}u_k$. Express the iterates of gradient descent applied to $g(u)$ in terms of $\hat{x_k}$, f and $M := P^T P$. This is often referred to as "preconditioned gradient descent". The choice of preconditioner gives quite a bit of flexibility in designing gradient based optimization algorithms, and can very significantly speed up convergence.

i. Expressing Gradient Descent for $g(u)$:
Gradient descent applied to $g(u)$ is given as $u_{k+1} = u_k - \alpha_k \nabla g(u_k)$. Remember that $\nabla g(u) = \nabla f(P^{-1}u)P^{-1}$. Substituting this into the gradient descent update for u_{k+1}, we have that $u_{k+1} = u_k - \alpha_k \nabla f(P^{-1}u_k)P^{-1}$.

ii. Because $\hat{x_k} = P^{-1}u_k$, we can manipulate the equation as $u_k = P\hat{x}_k$. Use the fact that $u_k = P\hat{x}_k$ for u_{k+1}:
$$P\hat{x}_{k+1} = P\hat{x}_k - \alpha_k \nabla f(P^{-1}(P\hat{x}_k))P^{-1}$$
Simplify $P^{-1}(P\hat{x}_k) = \hat{x}_k$ so that we have
$$P\hat{x}_{k+1} = P\hat{x}_k - \alpha_k \nabla f(\hat{x}_k)P^{-1}$$

iii.
$$\hat{x}_{k+1} = \hat{x}_k - \alpha_k P^{-1} \nabla f(\hat{x}_k)P^{-1} \qquad \text{(Multiplying through by } P^{-1} \text{ on the left)}$$

iv. Implementing our Preconditioner M:
Remember that $M = P^T P$. The term $P^{-1}\nabla f(\hat{x}_k)P^{-1}$ can be interpreted as a preconditioned gradient update:
$$\hat{x}_{k+1} = \hat{x}_k - \alpha_k M^{-1} \nabla f(\hat{x}_k)$$
Note that M adjusts the scaling of the gradient $\nabla f(\hat{x}_k)$ and this flexibility can help in cases of appropriate scaling and optimization landscapes.

v. Obtaining our final form:
The iterates of gradient descent applied to $g(u)$ in terms of $\hat{x}_k$, f, and M is:
$$\hat{x}_{k+1} = \hat{x}_k - \alpha_k M^{-1} \nabla f(\hat{x}_k)$$

Many of the modern optimization algorithms (ADAM, RMSProp, etc.) used in deep learning are (loosely) based around (diagonally) preconditioning stochastic gradient descent, which you'll learn about later in the quarter. Heuristically, the idea is that if the gradient of your objective varies more quickly (slowly) in some directions than others, you want to take relatively smaller (larger) steps in these directions.

2. A function f is *strictly convex* if

$$f(\lambda x + (1-\lambda)y) < \lambda f(x) + (1-\lambda)f(y), \quad \lambda \in (0,1).$$

(a) Give an example of a strictly convex function that does not have a minimizer.

Proof. Let $f : (-1, 0)$ be defined as $f(x) = x^2$. Since $\lambda, (1-\lambda) \in (0,1)$, we have that $\lambda^2 < \lambda$ and $(1-\lambda)^2 < (1-\lambda)$.
Therefore, we have that:

$$\begin{aligned} f(\lambda x + (1-\lambda)y) &= \lambda^2 x^2 + (1-\lambda)^2 y^2 \\ &< \lambda x^2 + (1-\lambda)y^2 && \text{(As } \lambda^2 < \lambda \text{ and } (1-\lambda)^2 < (1-\lambda)\text{)} \\ &= \lambda f(x) + (1-\lambda)f(y) && \text{(Based on our definition of } f\text{)} \end{aligned}$$

This proves that f is strictly convex. Additionally, for every $x \in (-1, 0)$, we have $f(x) = x^2$. Furthermore, $x^2 > 0$. Thus, we have that $f(x) > 0$. Since $f(x) > 0$ for all $x \in (-1, 0)$ and the domain of f is the open interval $(-1, 0)$, $f(x)$ does not attain its minimum value on $(-1, 0)$.

To see why, consider the infimum of $f(x)$ over $(-1, 0)$:

$$\inf_{x \in (-1,0)} f(x) = \lim_{x \to 0^-} x^2$$

As we know $\lim_{x \to 0^-} x^2 = 0$. Thus,

$$\inf_{x \in (-1,0)} f(x) = 0$$

However, $f(x) = 0$ is not actually achieved for any $x \in (-1, 0)$ as $0 \notin (-1, 0)$. Thus, $f(x)$ is strictly convex but does not have a minimizer on its domain. □

(b) Show that a sum of a strictly convex function and a convex function is strictly convex.

Proof. Suppose f is strictly convex and g is convex. Then, for all $\lambda \in (0, 1)$,

$$f(\lambda x + (1-\lambda)y) < \lambda f(x) + (1-\lambda)f(y)$$

and

$$g(\lambda x + (1-\lambda)y) \leq \lambda g(x) + (1-\lambda)g(y)$$

Thus,

$$\begin{aligned} (f+g)(\lambda x + (1-\lambda)y) &= f(\lambda x + (1-\lambda)y) + g(\lambda x + (1-\lambda)y) \\ &< \lambda f(x) + (1-\lambda)f(y) + \lambda g(x) + (1-\lambda)g(y) \\ &= \lambda(f+g)(x) + (1-\lambda)(f+g)(y) \end{aligned}$$

This proves $f + g$ is strictly convex. □

3. A function f is β-smooth when its gradient is β-Lipschitz continuous.

   (a) Find a global bound for β of the least-squares objective $\frac{1}{2}\|Ax-b\|^2$.

   First, let's start by calculating $\nabla(\frac{1}{2}||Ax-b||^2)$:

$$\nabla\left(\frac{1}{2}||Ax-b||^2\right) =$$
$$\nabla\left(\frac{1}{2}(Ax-b)^T(Ax-b)\right) =$$
$$\nabla\left(\frac{1}{2}(x^TA^TAx-2b^TAx+b^Tb)\right) =$$
$$A^TAx-A^Tb$$

   Thus, $\nabla(\frac{1}{2}||Ax-b||^2) = A^TAx-A^Tb$ Additionally, observe that A^TAx-A^Tb is continuous. Moreover, for any x, y:

$$\|(A^TAx-A^Tb)-(A^TAy-A^Tb)\| = \|A^TA(x-y)\| \leq \|A^TA\|\,\|x-y\|$$

   Therefore, the least squares objective $\frac{1}{2}\|Ax-b\|^2$ is β smooth provided:

$$\beta \geq \|A^TA\| = \lambda_{\max}(A^TA) = \sigma_{\max}(A)^2$$

   (b) Find a global bound for β of the regularized logistic objective

$$\sum_{i=1}^{n}\log(1+\exp(\langle a_i, x\rangle)) + \frac{\lambda}{2}\|x\|^2.$$

   Let $f(x) = G(Ax) + \frac{\lambda}{2}\|x\|^2$, where $G(z) = \sum_i g(z_i)$ and $g(z_i) = \ln(1+\exp(z_i))$.

   Then,

$$\nabla f(x) = A^T\nabla G(Ax) + \lambda x$$

   Now, observe that

$$(\nabla G(z))_i = g'(z_i) = \frac{\exp(z_i)}{1+\exp(z_i)} \in (0,1)$$

   Let $h(t) = e^t/(1+e^t)$ so that $h'(t) = e^t/(1+e^t)^2$. This is maximal at $t=0$ with value $h'(0) = \frac{1}{4}$. Therefore, $h(t)$ is Lipschitz with constant $\frac{1}{4}$.

   We now have that

$$\begin{aligned}\|\nabla G(u)-\nabla G(v)\|^2 &= \sum_{i=1}^{n}((\nabla G(u))_i - (\nabla G(v))_i)^2 \\ &\leq \sum_{i=1}^{n}\frac{1}{4}(u_i-v_i)^2 \\ &= \frac{1}{4}\|u-v\|^2\end{aligned}$$

   Therefore, $\nabla G : \mathbb{R}^n \to \mathbb{R}^n$ is Lipschitz with constant $\frac{1}{2}$. Therefore,

$$\|\nabla G(Ax)-\nabla G(Ay)\| \leq \|Ax-Ay\| \leq \frac{1}{2}\,\|A\|\,\|x-y\|$$

Then, by Triangle Inequality:

$$\begin{aligned}
\|\nabla f(x) - \nabla f(y)\| &= \|A^T \nabla G(Ax) + \lambda x - (A^T \nabla G(Ay) + \lambda y)\| \\
&= \|A^T (\nabla G(Ax) - \nabla G(Ay)) + \lambda (x - y)\| \\
&\leq \frac{1}{2} \|A^T\| \, \|\nabla G(Ax) - \nabla G(Ay)\| + \lambda \|x - y\| \\
&\leq \left(\frac{1}{2} \|A^T\| \, \|A\| + \lambda \right) \|x - y\|
\end{aligned}$$

This proves that f is β smooth for any

$$\beta \geq \frac{1}{2} \|A^T\| \, \|A\| + \lambda = \frac{1}{2} \sigma_{\max}(A)^2 + \lambda$$

(c) Do the gradients for Poisson regression admit a global Lipschitz constant?
The gradient of the Poisson regression objective does not admit a global Lipschitz constant because the term $\exp(\langle a_i, x \rangle)$ grows exponentially with $\|x\|$. This causes the gradient to be able to grow without a bound, which violates the Lipschitz condition. Thus, the gradients for Poisson regression do not admit a global Lipschitz constant.

4. I will complete the coding homework starting with the notebook uploaded to Canvas.