Name: Hongda Li Class: CSE 546 HW2B

B.1

Objective: Given the definition for the L2, L1 and the Infinity norm of real vector, show that $\|x\|_\infty \leq \|x\|_2 \leq \|x\|_1$.
First we are going to show that $\|x\|_2^2 \leq \|x\|_1^2$, starting from the definition of the norms we have:

$$\begin{aligned}
\|x\|_1^2 &= \left(\sum_{i=1}^n |x_i|\right)^2 \\
&= \sum_{i=1}^n \left(|x_i| \sum_{j=1}^n |x_j|\right) \\
&= \sum_{i=1}^n \left(|x_i|^2 + |x_i| \sum_{j=1, j\neq i}^n |x_j|\right) \\
&= \sum_{i=1}^n |x_i|^2 + \sum_{i=1}^n |x_i| \left(\sum_{j=1, j\neq i}^n |x_j|\right) \\
&= \|x\|_2^2 + \underbrace{\sum_{i=2}^n \sum_{j=1}^{i-1} 2|x_i||x_j|}_{\geq 0} \\
&\implies \|x\|_2^2 \leq \|x\|_1^2
\end{aligned} \tag{B.1.1}$$

And now we are going to shoe that $\|x\|_\infty^2 \leq \|x\|_2^2$. By the definition of the infinity norm, we know that therde exists $1 \leq m \leq n$ such that $x_m = \|x\|_\infty = \max_{1\leq i\leq n}(x_i)$. Then it can be said that:

$$\begin{aligned}
x_m^2 \leq & x_m^2 + \underbrace{\sum_{i=1, i\neq m}^n x_i^2}_{\geq 0} \\
x_m^2 = & \|x\|_\infty \leq \sum_{i=1}^n x_i^2 = \|x\|_2^2
\end{aligned} \tag{B.1.2}$$

And then combing together, we can take the square root because the function $\sqrt{\bullet}$ is monotone increase, hence it preserves the inequality, which will give us $\|x\|_\infty \leq \|x\|_2^2 \leq \|x\|_1$.

B.2

B.2.a

Objective: The function $\|x\|$ is a convex function.

$$\begin{aligned}
\|\lambda x + (1-\lambda)y\| &\leq \|\lambda x\| + \|(1-\lambda)y\| \\
&= \lambda\|x\| + (1-\lambda)\|y\|
\end{aligned} \tag{B.2.a.1}$$

Note, I just directly apply the Triangular inequality of the norm to get the inequality, and then becaues $\lambda \in [0, 1]$, so there is no absolute value, and notice that the resulting expression is the definition of Convexity the given function.

B.2.b

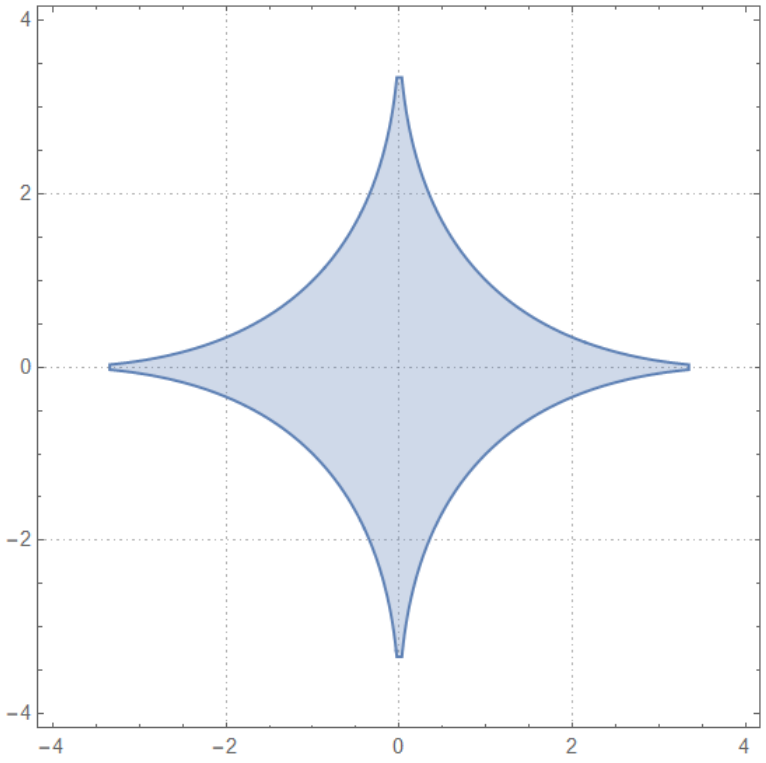
Objective: Show that the set $\{x \in \mathbb{R}^n : \|x\| \leq 1\}$ is a convex set. Let the set be denoted as S Let's take any 2 points in the set like $x \in S$, $y \in s$, then $\|x\| \leq 1$ and $\|y\| \leq 1$ for any line defined by the 2 points:

$$\begin{aligned} \|\lambda x + (1-\lambda)y\| &\leq \lambda \underbrace{\|x\|}_{\leq \lambda} + \underbrace{(1-\lambda)\|y\|}_{\leq 1-\lambda} \\ \implies \|\lambda x + (1-\lambda)y\| &\leq 1 \\ \implies \lambda x + (1-\lambda)y &\in S \end{aligned} \tag{B.2.b.1}$$

The first by the inequality of norm, and the second is by the definition of the fact that $x, y \in S$, and the third is by the definition of the set S.

B.2.c

The set $\{(x_1, x_2) : g(x_1, x_2) \leq 4\}$ domain of the function such that the value of the function is bounded by a given quantity. This is the unit norm ball defined by $p = \frac{1}{2}$. As we showned in lecture, it's not convex, and it looks like a star[1]:



B.3

B.3.a

Objective: Showing that, the squared loss function regularized with Eclidean norm (not squared) is convex. The trick is to show that, the squared loss function is convex, and we have shown the the Euclidean norm is convex back in B.2.a. And by showing that the positive weighted sum of 2 convex function is convex, we will be able to show that it's convex.
The loss function is convex because: $l_i(w) = (y_i - w^T x_i)^2$, this function is qaudratic, and it's second derivative

[1] Drawn by mathematica.

is diag(2) (The hessian fof the function wrt to w), which is a positive quantity. When the second derivative of a function is always larger than or equals to zero, the function is convex.
Here, we will show that the sum of 2 convex function is convex too. Let $f(x), g(x)$ be 2 convex function mapping $x \in \mathbb{R}^n$ to $\mathbb{R}$. Choose any 2 points $x, y \in \mathbb{R}^n$ then we have:

$$\begin{aligned} f(\lambda x + (1-\lambda)y) \leq & \lambda f(x) + (1-\lambda) f(y) \qquad \text{(B.3.a.1)} \\ & \wedge \\ g(\lambda x + (1-\lambda)y) \leq & \lambda g(x) + (1-\lambda) g(y) \\ \implies f(\lambda x + (1-\lambda)y) + g(\lambda x + (1-\lambda)y) & \leq \lambda f(x) + (1-\lambda) f(y) + \lambda g(x) + (1-\lambda) g(y) \\ f(\lambda x + (1-\lambda)y) + g(\lambda x + (1-\lambda)y) & \leq \lambda (f(x) + g(x)) + (1-\lambda)(f(x) + g(x)) \end{aligned}$$

It's not hatd to see that, if we set $h(x) := g(x) + h(x)$ then the last line of the statement will be the convexity of $h(x)$, proving the that sume of the 2 functions are still convex.
Note: If the sum 2 functions are still convex, then **the sum finite many function is still going to be convex**, this can be proved inductively, because the + sign is associatitve.
Using this, and using the fact that $\sum_{i=1}^{n} l_i(w) + \lambda \|w\|$ is convex, because all the function we are summing up are convex function.

B.2.b

Convex loss function is easier to optimize, it has minimal, and it's a convex set of minimal as well. And the use of a **strictly convex regularizer** will give **unique global optimal**, the strongly convex loss function is the best, because gradient descent performs really well on them.

B.4: Multinomial Logistic Regression

B.4.a

We are going to take the gradient on the loss function. Our objective is to find the log MLE of the following Loss Function:

$$\mathcal{L}(W) = -\sum_{i=1}^{n} \sum_{l=1}^{k} \mathbf{1}\{y_i = l\} \log \left(\frac{\exp\left(w^{(l)} \cdot x_i\right)}{\sum_{j=1}^{k} \exp\left(w^{j} \cdot x_i\right)} \right) \qquad \text{(B.4.1)}$$

To find the gradient of the loss function wrt the parameter matrix $W = [w^{(1)}, w^{(2)}, \cdots, w^{(k)}]$. We are going to take the gradient on the loss function wrt each column of the matrix and then we will stack then horizontal together to get the resulting matrix. Let's consider

$$\nabla_{w^{(m)}} L(W) = -\sum_{i=1}^{n} \sum_{l=1}^{k} \mathbf{1}\{y_i = l\} \nabla_{w^{(m)}} \left[\log \left(\frac{x_i^T \exp(x_i^T w^{(l)})}{\sum_{j=1}^{k} \exp(x_i^T w^{(j)})} \right) \right] \qquad \text{(B.4.2)}$$

Let's take a look at the inner part more carefully, and simpifying it will give us something like:

$$\begin{aligned} \nabla_{w^{(m)}} \left[\log \left(\frac{x_i^T \exp(x_i^T w^{(l)})}{\sum_{j=1}^{k} \exp(x_i^T w^{(j)})} \right) \right] &= \left(\begin{cases} x_i & l = m \\ 0 & \text{else} \end{cases} \right) - \partial_{w^{(m)}} \left[\log \left(\sum_{j=1}^{k} \exp\left(x_i^T w^{(j)}\right) \right) \right] \qquad \text{(B.4.3)} \\ &= \left(\begin{cases} x_i & l = m \\ 0 & \text{else} \end{cases} \right) - \frac{\partial_{w^{(m)}} \left[\sum_{j=1}^{k} \exp\left(x_i^T w^{(j)}\right) \right]}{\sum_{j=1}^{k} \exp\left(x_i^T w^{(j)}\right)} \\ &= \left(\begin{cases} x_i & l = m \\ 0 & \text{else} \end{cases} \right) - \frac{\exp(x_i^T w^{(m)}) x_i}{\sum_{j=1}^{k} \exp\left(x_i^T w^{(j)}\right)} \\ &= x_i \left(\mathbf{1}\{l = m\} - \frac{\exp(x_i^T w^{(m)}) x_i}{\sum_{j=1}^{k} \exp\left(x_i^T w^{(j)}\right)} \right) \end{aligned}$$

Do take note that $l = m \iff y_i = l$ and these 2 evets happens at the same time and they are interchangable.
And we follow the tradition of using the label vector from previous HW1, which is that each of the y_i is a column vector, and it's 1 in the "i" th position of the vector, denoting the the data x_i has label i.
Another thing we are going to make use is the softmax(θ) which is a vector function mapping from $\mathbb{R}^k$ to $\mathbb{R}^k$:

$$(\text{softmax}(\theta))_m = \frac{\exp(\theta_m)}{\sum_{j=1}^k \exp(\theta_j)}$$

Where, we showcase the m th element of this function when we put into the vector θ.
We may use this to simply simpify the cases situation above, and that should give us:

$$\begin{aligned} -\sum_{i=1}^n \mathbf{1}\{y_i = l\} \nabla_{w^{(m)}} \left[\log \left(\frac{x_i^T \exp(x_i^T w^{(l)})}{\sum_{j=1}^k \exp(x_i^T w^{(j)})} \right) \right] &= -\sum_{i=1}^n x_i \left(\mathbf{1}\{y_i = l\} - \underbrace{\frac{\exp(x_i^T w^{(m)})}{\sum_{j=1}^k \exp\left(x_i^T w^{(j)}\right)}}_{(\text{softmax}((x_i^T W)^T))_m} \right) \\ &= -\sum_{i=1}^n x_i \left(\mathbf{1}\{y_i = l\} - (\text{softmax}(W^T x_i))_m \right) \end{aligned} \tag{B.4.4}$$

Now, I am going horizontally stack all these vector together (as m goes from 1 to d.), and notice that it can be written as an outter product, and the condition $\mathbf{1}(y_i = l)$ can be replaced with the appropriate label vector y_i, and this will be like:

$$\nabla_W \mathcal{L}(W) = -\sum_{i=1}^n x_i (y_i - \text{softmax}(W^T x_i))^T \tag{B.4.5}$$

Then in this case, we just need to make the definition that:

$$\hat{y}_i^W = \text{sfotmax}(W^T x_i)$$

Then substituting this back to the previous part, we have the expression that:

$$\nabla_W \mathcal{L}(W) = -\sum_{i=1}^n x_i (y_i - \hat{y}_i^{(W)})^T \tag{B.4.6}$$

B.4.b

From the previous HW1, it had been shown that the gradient of the Linear regression. But it's not exactly the same be cause last time there is a Ridge Regularizer that we need to remove in order to let it work this time.
The loss function is a scalar function, and we we are taking wrt to $W_{i,j}$, the element in the ith row and jth column of the weight matrix.
So here I am going to use the idea of automatic differentiation to figure out the gradient wrt to the weight matrix. It's going to be like:

$$\begin{aligned} \partial_{W[i,j]} \left[\frac{1}{2} \sum_{k=1}^n \|y_i - W^T x_i\|_2^2 \right] &= \frac{1}{2} \sum_{k=1}^n \left(\partial_{W[i,j]} [W^T x_k] \right)^T (y_k - W^T x_k) \\ &= \sum_{k=1}^n (x_k[i] \vec{e}_j)^T (y_k - W^T x_k) \\ &= \sum_{k=1}^n x_k[i] (y_k - W^T x_k)[j] \end{aligned} \tag{B.4.b.1}$$

And the key here, is that $\partial_{W[i,j]}[W^T x_k]$ equals to $x_k[i]\hat{e_j}$. So here, we are taking the derivative on a vector wrt to a scalar.

Only the j th output of the vector $W^T x_k$ is related to $W_{i,j}$ because $W_{i,j}$ is at the j th column of W^T, and the element is multiplied by x_i because W_i, j is at the j th column of W^T, therefore, the output is a sparse vector that can be represented by $x_k[i]\vec{e}_j$ where $\vec{e}$ is the j th standard basis vector.
Lastly, I wanted to point out that, if we sweep through the rows and columns of the matrix, we get the outter product of these 2 indexed vector, hence we have:

$$\nabla_W \left[\frac{1}{2} \sum_{k=1}^{n} \|y_i - W^T x_i\|_2^2 \right] = \sum_{k=1}^{n} x_k (y_k - W^T x_k)^T \tag{B.4.b.2}$$

With the susbtitution that $\tilde{y}_i^{(W)} = W^T x_i$, we have:

$$\nabla_W \left[\frac{1}{2} \sum_{k=1}^{n} \|y_i - W^T x_i\|_2^2 \right] = \sum_{k=1}^{n} x_k (y_k - \tilde{y}_i^{(W)})^T \tag{B.4.b.3}$$

And notice that it has the exact same form as the Multinomial Logistic Regression from before.

B.4.c

B.5: Confidence Interval of Least Squares Estimation: Bounding Estimate