CPSC 488/588 AI Foundation Models
Fall 2023
HW 1

Instructions: copy this project at `https://www.overleaf.com/read/gjvcgrzjyxfm`, complete the solutions, and return your solutions in pdf format.

Full Name: Sneha
Netid: ss3993

1. **Q1**

   Warmup. this part is about refreshing your calculus on calculating derivatives of functions.

   (a) Given the function:

   $$f(x) = \sin(3x^2 + 4\cos(x))$$

   find $\partial f/\partial x$ using the chain rule.

   Solution:

   $$\frac{\partial f}{\partial x} = \cos(3x^2 + 4\cos(x)) \cdot (6x - 4\sin(x))$$

   (b) Given the function:

   $$f(x) = e^{\tanh(2x^3 - 5x^2 + x)}$$

   find $\partial f/\partial x$ using the chain rule.
   Recall: $\frac{\partial \tanh(x)}{\partial(x)} = 1 - \tanh^2(x) = \text{sech}^2(x)$

   Solution:

   $$\frac{\partial f}{\partial x} = e^{\tanh(2x^3 - 5x^2 + x)} \cdot \text{sech}^2(2x^3 - 5x^2 + x) \cdot \left(6x^2 - 10x + 1\right)$$

   (c) Consider the function:

   $$f(x, y, z) = x^2 \sin(yz) + e^{yz} - z^3 y$$

   find partial derivatives of the function with respect to each variable separately.

   Solution:

   $$\frac{\partial f}{\partial x} = 2x\sin(yz),$$
   $$\frac{\partial f}{\partial y} = x^2 z\cos(yz) + ze^{yz} - z^3,$$
   $$\frac{\partial f}{\partial z} = x^2 y\cos(yz) + ye^{yz} - 3z^2 y.$$

2. **Q2 Matrix calculus.** Recall that matrices are a way of organizing data into rows and columns. Answer the following questions:

   (a) Consider a function given by:

   $$f(\mathbf{x}) = \mathbf{c}^T\mathbf{A}\mathbf{x} \qquad (1)$$

   where

   $$\mathbf{x} = \begin{bmatrix} x_0 \\ x_1 \\ x_2 \end{bmatrix}$$

is a column vector of variables,

$$\mathbf{c} = \begin{bmatrix} 2 & 3 & 1 \end{bmatrix}$$

is a constant vector, and

$$\mathbf{A} = \begin{bmatrix} a_{00} & a_{01} & a_{02} \\ a_{10} & a_{11} & a_{12} \\ a_{20} & a_{21} & a_{22} \end{bmatrix}$$

is a integer valued matrix,
Derive the gradient of f with respect to $\mathbf{x}$.

Solution: Given the function

$$f(x) = \begin{bmatrix} 2 & 3 & 1 \end{bmatrix} \begin{bmatrix} a_{00} & a_{01} & a_{02} \\ a_{10} & a_{11} & a_{12} \\ a_{20} & a_{21} & a_{22} \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ x_2 \end{bmatrix}$$

we can expand Ax as:

$$Ax = \begin{bmatrix} a_{00}x_0 + a_{01}x_1 + a_{02}x_2 \\ a_{10}x_0 + a_{11}x_1 + a_{12}x_2 \\ a_{20}x_0 + a_{21}x_1 + a_{22}x_2 \end{bmatrix}$$

The function $f(x)$ expands to:

$$f(x) = 2a_{00}x_0 + 3a_{10}x_0 + a_{20}x_0 + 2a_{01}x_1 + 3a_{11}x_1 + a_{21}x_1 + 2a_{02}x_2 + 3a_{12}x_2 + a_{22}x_2$$

Differentiating with respect to the components of x, we get:

$$\frac{\partial f(x)}{\partial x_0} = 2a_{00} + 3a_{10} + a_{20}$$
$$\frac{\partial f(x)}{\partial x_1} = 2a_{01} + 3a_{11} + a_{21}$$
$$\frac{\partial f(x)}{\partial x_2} = 2a_{02} + 3a_{12} + a_{22}$$

(b) Given the function

$$f = \mathbf{x}^\top \cdot \mathbf{A} \cdot \mathbf{x} + c \cdot \sin(\mathbf{y})^\top \cdot \mathbf{x}$$

where

- A is a symmetric matrix
- c is a scalar
- x is a vector
- y is a vector

Derive the gradients with respect to $\mathbf{x}$ and $\mathbf{y}$.

Solution:
Gradient with respect to x:
Using the identity $\nabla_x(x^T Ax) = (A+A^T)x$ and we also know that A is symmetric (i.e., $A = A^T$), so the derivative can be simplified to $2Ax$.

$$\frac{\partial f}{\partial x} = 2Ax + c\sin(y)$$
$$\frac{\partial f}{\partial y} = cx\cos(y)$$

(c) Given the function

$$g = \mathbf{x}^\top \mathbf{B}\mathbf{y} + d\tanh(\mathbf{z})^\top \mathbf{x}$$

where

- $\mathbf{B}$ is an arbitrary matrix
- d is a scalar
- $\mathbf{x}$ is a vector
- $\mathbf{y}$ is a vector of the same dimension as $\mathbf{x}$
- $\mathbf{z}$ is a vector

Derive the gradients with respect to $\mathbf{x}$, $\mathbf{y}$, and $\mathbf{z}$.

Hint: recall $\frac{\partial \tanh(x)}{\partial(x)} = 1 - \tanh^2(x) = \text{sech}^2(x)$

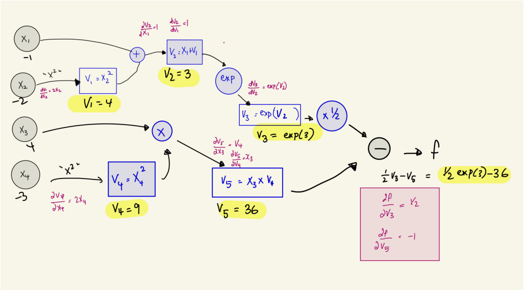
Solution:

1. Gradient with respect to x:
$$\frac{\partial g}{\partial x} = By + d\tanh(z)$$
2. Gradient with respect to y:
$$\frac{\partial g}{\partial y} = B^T x$$
3. Gradient with respect to z:
$$\frac{\partial g}{\partial z} = d \cdot \text{sech}^2(z) \cdot x$$

3. **Q3 Automatic differentiation**

(a) Consider the following function: $f(x_1, x_2, x_3, x_4) = \frac{1}{2}\exp(x_1 + x_2^2) - (x_3 * x_4^2)$

i. draw the computation graph corresponding to this function and fill in the gradient values for all of the intermediate nodes and the leaves in the computation graph. Assume the values for x_1, x_2, x_3, x_4 are $-1, 2, 4, -3$ respectively.



ii. **Solution:**

iii. Derive the gradients of f with respect to its inputs ∇f using symbolic differentiation and chain rule.

Solution:

$$\begin{aligned}
\frac{\partial f}{\partial x_1} &= \frac{1}{2}\exp(x_1 + x_2^2) = 1/2 * exp(3) = exp(3)/2 \\
\frac{\partial f}{\partial x_2} &= x_2 \exp(x_1 + x_2^2) = 2 * exp(3) \\
\frac{\partial f}{\partial x_3} &= -x_4^2 = -9 \\
\frac{\partial f}{\partial x_4} &= -2x_3x_4 = 24
\end{aligned}$$

4. **Q4** Explain the reason behind using negative sampling in the SkipGram word embeddings model.

Solution: In a naive implementation of the SkipGram model, for each training example, we would update weights for all words in the vocabulary using softmax. By using negative sampling, instead of computing the softmax over all words in the vocabulary, we only need to compute it for the actual context word and a small set of negative samples. This greatly reduces the computational burden.Furthermore, negative sampling indirectly also sends implicit negative information which as a result the model implicitly gains information about what words are unlikely to appear in the context improving word vector quality.

5. **Q5 transformer**

   (a) In the multi-head self-attention operation, what is the cost of computation (in terms of number of number of FLoating point OPerations)? Assume b is batch size, m is sequence length, d is the model dimension, and h is the number of attention heads. Assume the dimensionality of keys and queries are $d/2h$.

   Note: You need to derive the answer, just providing the final answer is not sufficient.

Solution:

Scaled Dot Product Attention:
Dot product for one head: b $\times m \times m \times \frac{d}{2h}$
For h heads:$b \times m \times m \times d = bm^2d$

Applying Softmax:
Softmax operations: b $\times m \times m \times h \approx bm^2h$

Computing the weighted average:
For one head: b $\times m \times m \times \frac{d}{2h}$
For h heads:$b \times m \times m \times d = bm^2d$

Output projection:
Output operations: 2 $\times b \times m \times d \times d = 2bm^2d$

Total FLOPs:
Total operations: 3bmd$^2 + bm^2d + bm^2h + bm^2d + 2bm^2d = 4bmd^2 + 3bm^2d + bm^2h$

(b) What is the cost of computation in terms of number of FLoating point Operations for multi-head attention in the backward pass, when the model is being trained? (use the same assumptions as the above questions).

Solution:

$$\begin{aligned}
\textbf{1. Projection to Q, K, V: } & 6bmd^2 \\
\textbf{2. Scaled Dot Product Attention: } & 2bm^2d \\
\textbf{3. Applying Softmax: } & bm^2h \\
\textbf{4. Computing the weighted average: } & 2bm^2d \\
\textbf{5. Output projection: } & 4bm^2d \\
\hline
\textbf{Total Backward FLOPs: } & 6bmd^2 + 9bm^2d + bm^2h
\end{aligned}$$

(c) What is the cost of computation in terms of number of FLoating point Operations for **Grouped Query Attention** where $G = k/4$ is the number of groups?

Solution:

$$\begin{aligned}
\textbf{1. Projection to Q, K, V: } & 3bmd^2 \\
\textbf{2. Grouping Queries and Dot Product: } & \frac{bm^2d}{2} \\
\textbf{3. Applying Softmax: } & bm^2h \\
\textbf{4. Computing the weighted average: } & \frac{bm^2d}{2} \\
\hline
\textbf{Total FLOPs for Grouped Query Attention: } & 4bmd^2 + bm^2h
\end{aligned}$$