DL HW1

Mehdi Jamalkhah

July 2024

1 Partial Derivative

- $$y_i = \sum_{k=1}^{n} a_{ik}x_k \Rightarrow \frac{\partial y_i}{\partial x_j} = a_{ij} \Rightarrow \frac{\partial y}{\partial x} = A$$

- $$\frac{\partial y_i}{\partial z_j} = \sum_{k=1}^{n} a_{ik}\frac{\partial x_k}{\partial z_j} = (A\frac{\partial x}{\partial z})_{ij} \Rightarrow \frac{\partial y}{\partial z} = A\frac{\partial x}{\partial z}$$

- $$\begin{aligned} \alpha &= \sum_{i=1}^{m}\sum_{j=1}^{n} y_i a_{ij} x_j \\ \frac{\partial \alpha}{\partial x_j} &= \sum_{i=1}^{m} y_i a_{ij} = (y^T A)_j \Rightarrow \frac{\partial \alpha}{\partial x} = y^T A \\ \frac{\partial \alpha}{\partial y_j} &= \sum_{j=1}^{n} a_{ij} x_j = (x^T A^T)_j \Rightarrow \frac{\partial \alpha}{\partial y} = x^T A \end{aligned}$$

- $$\begin{aligned} \alpha &= \sum_i y_i x_i \\ \frac{\partial \alpha}{\partial z_j} &= \sum_i x_i \frac{\partial y_i}{\partial z_j} + y_i \frac{\partial x_i}{\partial z_j} = (x^T \frac{\partial y}{\partial z})_j + (y^T \frac{\partial x}{\partial z})_j \Rightarrow \frac{\partial \alpha}{\partial z} = x^T \frac{\partial y}{\partial z} + y^T \frac{\partial x}{\partial z} \end{aligned}$$

- $$\begin{aligned} AA^{-1} = I &\Rightarrow \frac{\partial A}{\partial \alpha}A^{-1} + A\frac{\partial A^{-1}}{\partial \alpha} = 0 \\ &\Rightarrow A\frac{\partial A^{-1}}{\partial \alpha} = -\frac{\partial A}{\partial \alpha}A^{-1} \Rightarrow \frac{\partial A^{-1}}{\partial \alpha} = -A^{-1}\frac{\partial A}{\partial \alpha}A^{-1} \end{aligned}$$

2 Hessian Matrix

$$\nabla\psi = \begin{bmatrix} \frac{\partial\psi}{\partial u} \\ \frac{\partial\psi}{\partial v} \\ \frac{\partial\psi}{\partial z} \end{bmatrix} \Rightarrow J = \begin{bmatrix} \frac{\partial^2\psi}{\partial u\partial u} & \frac{\partial^2\psi}{\partial v\partial u} & \frac{\partial^2\psi}{\partial z\partial u} \\ \frac{\partial^2\psi}{\partial u\partial v} & \frac{\partial^2\psi}{\partial v\partial v} & \frac{\partial^2\psi}{\partial z\partial v} \\ \frac{\partial^2\psi}{\partial u\partial z} & \frac{\partial^2\psi}{\partial v\partial z} & \frac{\partial^2\psi}{\partial z\partial z} \end{bmatrix} = H(y)$$

3 Avoiding Becoming Asymmetric

Some images have a symmetry in their structure, like a symmetric face. Now, if we use this knowledge and apply an inductive bias to our model to learn symmetric weights, it means that we need fewer parameters, something like weight sharing.

$$\begin{aligned}
&R(w) = w^T S w = w_1^2 s_{11} + w_2^2 s_{22} + w_1 w_2 (s_{12} + s_{21}) \\
&w_1 \to w_2 \Rightarrow s_{12} + s_{21} + s_{22} + s_{11} = 0 \\
&e.g. \quad S = \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}
\end{aligned}$$

4 Backpropagation Basics

- $W_1 \Rightarrow D_{a1} \times D_x \qquad b_1 \Rightarrow D_{a1} \times 1 \qquad W_2 \Rightarrow 2 \times D_{a1} \qquad b_2 \Rightarrow 2 \times 1$

- $$\frac{\partial J}{\partial \hat{y}^{(i)}} = -\frac{1}{m}\left(\frac{y^{(i)}}{\hat{y}^{(i)}} - \frac{1 - y^{(i)}}{1 - \hat{y}^{(i)}}\right) = \delta_1$$

- $$\frac{\partial \hat{y}^{(i)}}{\partial z_2} = (1 - \sigma(z_2))\sigma(z_2) = \delta_2$$

- $$\frac{\partial z_2}{\partial a_1} = W_2^T = \delta_3$$

- $$\frac{\partial a_1}{\partial z_1} = \begin{bmatrix} I(z_{11} > 0) & 0 & \dots & 0 \\ 0 & I(z_{12} > 0) & \dots & 0 \\ 0 & 0 & \ddots & 0 \\ 0 & 0 & \dots & I(z_{1,D_{a1}} > 0) \end{bmatrix} = \delta_4$$

- $$\frac{\partial z_1}{\partial W_{1ij}} = [0 \quad \dots \quad 0 \quad x_j \quad 0 \quad \dots \quad 0]$$

- $$\frac{\partial J}{\partial W_1} = \delta_1 \delta_2 \delta_3 \delta_4 \delta_5$$

5 Optimization

5.1 Beale Function

- The Beale function is a convex function, as it is the sum of three quadratic functions, and quadratic functions are inherently convex.

  Convex functions possess a unique global minimum, which means that when optimizing a convex function, we can be confident of finding the optimal solution by searching for the minimum value of the function. This property makes optimization tasks more straightforward and reliable, especially when employing gradient descent techniques, which may otherwise converge to a local minimum.

- $$\frac{\partial f}{\partial x_1} = 2(1.5 - x_1 + x_1x_2)(x_2 - 1) + 2(2.25 - x_1 + x_1x_2^2)(x_2^2 - 1) + 2(2.625 - x_1 + x_1x_2^3)(x_2^3 - 1)$$
  $$\frac{\partial f}{\partial x_2} = 2(1.5 - x_1 + x_1x_2)x_1 + 2(2.25 - x_1 + x_1x_2^2)(2x_1x_2) + 2(2.65 - x_1 + x_1x_2^3)(3x_2^2x_1)$$
  $$\nabla f(x^t = (0,1)) = (0,0) \Rightarrow x^{t+1} = (0,1)$$

5.2 Quadratic Function

$$\begin{aligned}
&h(x) = \frac{1}{2}x^TQx + x^Tc + b \Rightarrow \nabla_x h = Qx + c = Qx - Qx^* \\
&\text{update rule} : x^{k+1} = x^k - \alpha Q(x - x^*) \\
&\text{eigen decomposition} : Q = V\Lambda V^T \\
&\text{change of basis} : \hat{x}^k = V^T(x^k - x^*) \Rightarrow x^k = V\hat{x}^k + x^* \\
&V\hat{x}^{k+1} + x^* = V\hat{x}^k + x^* - \alpha Q(x - x^*) \\
&\hat{x}^{k+1} = \hat{x}^k - \alpha V^TQ(VV^T)(x - x^*) \\
&= \hat{x}^k - \alpha\Lambda\hat{x}^k = (1 - \alpha\Lambda)\hat{x}^k
\end{aligned}$$

By recursion rule we have

$$\hat{x}_i^k = (1 - \alpha\lambda_i)^k\hat{x}_i^0$$

Moving back to our original space x, we can see that

$$x^k - x^* = V\hat{x}^k = \sum_i \hat{x}_i^k v_i = \sum_i x_i^0(1 - \alpha\lambda_i)^k v_i$$

5.3 AdaMax

- $$v^{(k)} = max(\beta_2 v^{(k-1)}, (1 - \beta_2)|\partial_W E^{(k)}|)$$
- This technique can be useful in situations where the gradients are very sparse or have a high variance. It is also more robust to noisy data compared to Adam, since it does not rely on the full history of past gradients.