## Question 4

**1. Explain the basic principle behind gradient descent. Why this technique is core in machine learning?**

The gradient descent is a optimization technique, by optimization technique we mean it updates the model parameters given a function of such parameter iteratively. the explanation is much easier if we start by mentioning the equation

$$\theta' = \theta - \eta \nabla C_\theta$$

now if we imagine the overall solution space as a subset of $\mathbb{R}^3$ and remembering that $\nabla C_\theta$ is a vector aligned with the direction of largest change in the dimension $\theta$. The minus sign implies an update to the value of the parameter in the opposite direction. Note that we expect the positive direction to point away from a global minimum thus increasing the training loss, that's the reason for the minus sign.

This reason why this technique is core in machine learning is due to most mainstream optimizations being variations of the gradient descent.

**2. Explain the distinctions between batch, mini-batch and stochastic gradient descent, make sure to compare advantages and disadvantages of each.**

SGD (stochastic gradient descent) estimates the value of the gradient after each sample

$$\theta = \theta' - \eta \nabla C_{\theta_i}, i \in \mathbb{R}^d$$

notable advantage is reduction in training time, however training performance might be impacted however due to the noise the algorithm might get unstuck from saddle or local minima points.

Batch gradient descent is essential the technique described in the previous question with the modification in how we compute the final gradient

$$\theta = \theta' - \frac{\eta}{n} \Sigma_i \nabla C_{\theta_i}, i \in \mathbb{R}^d$$

advantages is possibly a good accuracy, but a notable disadvantages is no tolerance for error if the algorithm get stuck in a saddle or local minima it might never "unstuck" thus possibility of no convergence might be high in some cases.

Mini batch gradient descent is a variation of Batch gradient descent where a subset of samples are selected and thus learning happens faster and also resistant to saddle and local minima the disadvantage is more fine tuning in the hyperparameters required and not as accurate as the Batch gradient descent.

**3. How the partial derivaties impact the model weights during training?**

already answered in the first question, just change the parameter from $\theta$ to $w_{ij}^l$ the j-th weight of the i-th neuron in the l-th layer.