Bonus Question for Lab 9:
Essay about Regularizations in Machine Learning

Mohammad Parsa Dini
Student ID: 400101204

Regularization Techniques in Machine Learning

Regularization is a set of techniques used to prevent overfitting in machine learning models by adding additional constraints or penalties to the model. Here are three common regularization methods, explained in detail with examples:

1. L2 Regularization (Ridge Regression)

L2 regularization adds a penalty equal to the sum of the squared values of the weights to the loss function. This encourages the model to keep the weights small, which helps prevent overfitting. The regularization term penalizes large weights, making the model less sensitive to variations in the training data.

$$\text{L2 Regularization: } \mathcal{L} = \mathcal{L}_{\text{original}} + \lambda \sum_{i=1}^{n} w_i^2$$

where λ is the regularization parameter, w_i are the weights, and $\mathcal{L}_{\text{original}}$ is the original loss function.

Example: Consider a linear regression problem where we want to fit a line to data points. Without regularization, the model might overfit, especially if there are outliers. By adding L2 regularization, we ensure that the model does not assign excessively high weights to any single feature, leading to a more generalized model.

2. L1 Regularization (Lasso Regression)

L1 regularization adds a penalty equal to the sum of the absolute values of the weights to the loss function. This can lead to sparse models, where some weights become exactly zero, effectively performing feature selection. By shrinking some coefficients to zero, L1 regularization simplifies the model.

$$\text{L1 Regularization: } \mathcal{L} = \mathcal{L}_{\text{original}} + \lambda \sum_{i=1}^{n} |w_i|$$

where λ is the regularization parameter, w_i are the weights, and $\mathcal{L}_{\text{original}}$ is the original loss function.

Example: Imagine we are using Lasso regression for feature selection in a high-dimensional dataset with many features. L1 regularization will shrink the coefficients of less important features to zero, effectively selecting only the most relevant features. This reduces the complexity of the model and can improve interpretability.

3. Dropout

Dropout is a technique where, during training, randomly selected neurons are ignored (dropped out). This means their contribution to the activation of downstream neurons is temporarily removed on the forward pass, and any weight updates are not applied to the neuron on the backward pass. This helps prevent the model from becoming too dependent on any single neuron and improves generalization.

$$\text{Dropout: } \hat{y} = f\left(\sum_{i \in \text{active neurons}} w_i x_i\right)$$

where f is the activation function, w_i are the weights, x_i are the inputs, and the summation is only over the active neurons.

Example: Consider a deep neural network for image classification. Without dropout, the network might overfit the training data, memorizing specific features. By applying dropout, we ensure that different subsets of neurons learn different representations, leading to a more robust model that generalizes better to unseen data.

4. Early Stopping

Early stopping is a form of regularization used to prevent overfitting by monitoring the model's performance on a validation set during training. When the performance stops improving, training is halted.

Example: In a deep learning model, we might train for 100 epochs. However, if the validation loss stops decreasing and starts increasing after 20 epochs, we can stop training early. This prevents the model from overfitting the training data and ensures better performance on the validation set.

5. Data Augmentation

Data augmentation involves increasing the diversity of the training dataset by applying random transformations such as rotations, flips, and shifts. This helps the model generalize better to new data.

Example: In image classification, applying random rotations, shifts, and flips to the training images can create new training examples. This helps the model become invariant to these transformations and improves its ability to generalize to new images.