Deep Learning - Foundations and Concepts

Chapter 9. Regularization

nonlineark@github

March 2, 2025

Outline

Inverse problems

- Most machine learning tasks are examples of inverse problems:
  - Forward problem: Given a conditional distribution $p(t|x)$ along with a finite set of input points $\{x_1, \ldots, x_N\}$, sample corresponding values $\{t_1, \ldots, t_N\}$ from that distribution.
  - Inverse problem: Infer a distribution given only a finite number of samples.
- We need a way to choose a specific distribution from amongst the infinitely many possibilities. The preference for one choice over others is called inductive bias or prior knowledge:
  - Small changes in the input values should lead to small changes in the output values: The sum-of-squares regularizer.
  - When detecting objects in images, there should be translation invariance: Convolutional neural network.
  - Additional data from a different, but related, task can be used to help determine the learnable parameters in a neural network: Transfer learning and multi-task learning.

No free lunch theorem

The no free lunch theorem: Every learning algorithm is as good as any other when averaged over all possible problems:

- Even very flexible neural networks contain important inductive biases, it is not possible to learn purely from data in the absence of any bias.
- In trying to find general-purpose learning algorithms, we are really seeking inductive biases that are appropriate to the broad classes of applications that will be encountered in practice.
- Inductive biases can be incorporated through:
  - The form of distribution.
  - The addition of a regularization term to the error function used during training.
  - The training process.

Symmetry and invariance

- In many applications of machine learning, the predictions should be unchanged under one or more transformations of the input variables:
  - Translation invariance.
  - Scale invariance.
- Transformations that leave particular properties unchanged represent symmetries. The set of all transformations corresponding to a particular symmetry form a group.

Symmetry and invariance

Efficient approaches for encouraging an adaptive model to exhibit the required invarances:

- Pre-processing: Invariances are built into a pre-processing stage by computing features of the data that are invariant under the required transformations.
- Regularized error function: A regularization term is added to the error function to penalize changes in the model output when the input is subject to one of the invariant transformations.
- Data augmentation: The training set is expanded using replicas of the training data points, transformed according to the desired invariances and carrying the same output target values as the untransformed examples.
- Network architecture: The invariance properties are built into the structure of a neural network through an appropriate choice of network architecture.

Equivariance

A generalization of invariance is called equivariance in which the output of the network, instead of remaining constant when the input is transformed, is itself transformed in a specific way:

$$\mathcal{S}(\mathcal{T}(I)) = \tilde{\mathcal{T}}(\mathcal{S}(I))$$

For example:

- $\mathcal{S}$: Measures the orientation of an object within an image.
- $\mathcal{T}$: Rotation.
- $\tilde{\mathcal{T}}$: Increase or decrease the scalar orientation.