INF721 - Deep Learning
L2: Machine Learning (v2)

Prof. Lucas N. Ferreira
Universidade Federal de Viçosa

2024/2

1 Introduction

Deep learning is a class of Machine Learning algorithms based on neural networks. Before diving into deep learning, it is essential to understand the basics of machine learning, including types of learning problems, algorithms, and evaluation techniques. This lecture provides an overview of machine learning concepts and sets the stage for exploring deep learning in subsequent lectures.

1.1 Definitions

- **Artificial Intelligence (AI):** The field of computer science focused on creating systems that can perform tasks requiring human intelligence.
- **Machine Learning:** A subset of AI that focuses on developing algorithms and models that allow computers to learn from and make predictions or decisions based on data.
- **Deep Learning:** A class of machine learning algorithms based on artificial neural networks with multiple layers.

1.2 Brief History of Machine Learning and AI

- 1940s: McCulloch and Pitts design the first artificial neurons (not learned)
- 1950s: Rosenblatt develops the perceptron, capable of learning linear problems
- 1960s: Minsky shows limitations of single-layer perceptrons (e.g., XOR problem)
- 1980s: Multi-layer perceptrons and backpropagation algorithm developed
- 2000s: Support Vector Machines (SVMs) gain popularity
- 2010s onwards: Deep learning breakthroughs and rapid advancements

2 Types of Machine Learning

Machine learning can be broadly categorized into three main types based on the nature of the learning process and the data available:

2.1 Supervised Learning

In supervised learning, the algorithm learns a function from labeled data. The dataset contains both input features $\mathbf{x}$ and corresponding target labels y.

Formal definition: Given a dataset $D = \{(\mathbf{x}^{(i)}, y^{(i)})\}_{i=1}^{m}$ where:

- $\mathbf{x}^{(i)} \in \mathbb{R}^d$ is the feature vector of the i-th example
- $y^{(i)}$ is the label or class of the i-th example
- m is the number of examples in the dataset
- d is the dimensionality of the feature vector

The goal is to learn a function $h : \mathbb{R}^d \to C$ that maps input features $\mathbf{x}$ to output labels y, where C is the the *label space*, i.e., the set of possible labels.

2.1.1 Types of Supervised Learning Problems

Supervised learning problems can be further classified based on the nature of the label space C:

1. **Classification:** The label space C is categorical (discrete).
   - Binary Classification: Two possible classes
     Example: Spam Detection
     - Input: Email content (text)
     - Output: Spam (1) or Not Spam (0)
     - Feature representation: Word frequency counts
   - Multi-class Classification: More than two classes
     Handwritten Digit Recognition:
     - Input: Image of a handwritten digit
     - Output: Digit class (0-9)
     - Feature representation: Pixel values
2. **Regression:** The labels space C is continuous.

   Example: House Price Prediction:

   - Input: House features (size, location, etc.)
   - Output: Predicted price (continuous value)
   - Feature representation: Tabular data

2.2 Unsupervised Learning

In unsupervised learning, the algorithm learns a function from unlabeled data. The dataset contains only input features $\mathbf{x}$ without corresponding target labels.

Formal definition: Given a dataset $D = \{\mathbf{x}^{(i)}\}_{i=1}^{m}$ where:

- $\mathbf{x}^{(i)} \in \mathbb{R}^d$ is the feature vector of the i-th example
- m is the number of examples in the dataset
- d is the dimensionality of the feature vector

The goal is to find patterns, structures, or relationships in the data without explicit labels.

2.2.1 Types of Unsupervised Learning Problems

1. **Clustering:** Group similar data points together.

   Example: Feature Compression

   - Input: High-dimensional data
   - Output: Lower-dimensional representation

2. **Dimensionality Reduction:** Reduce the number of features while preserving important information.

   Example: Customer Segmentation

   - Input: Customer behavior data
   - Output: Groups of similar customers

3. **Generative modelling:** Learn to generate new data similar to the training data.

   Example: Image Generation

   - Input: Large dataset of images
   - Output: New, synthetic images

2.3 Reinforcement Learning

Reinforcement learning involves an agent learning to make decisions by interacting with an environment. The agent receives feedback in the form of rewards or penalties for each action it takes.

Formal definition: Given a set states S and actions A, the goal is to learn a function $\pi : S \to A$ (called policy) that maximizes the expected sum of rewards.

Examples:

- Game-playing AI (e.g., AlphaGo)
- Robotic control systems

3 Data Types in Machine Learning

Understanding the types of data used in machine learning is crucial for selecting appropriate algorithms and preprocessing techniques.

3.1 Structured Data

Structured data is organized in a predefined format, typically in tables with rows and columns. Each column represents a specific attribute or feature.

Examples:

- Relational databases
- Spreadsheets
- CSV files

3.2 Unstructured Data

Unstructured data lacks a predefined format or structure. It requires more complex preprocessing and feature extraction techniques.

Examples:

- Text documents
- Images
- Audio files
- Video files

4 Supervised Learning Algorithms

In order to learn a function, supervised learning algorithm must define a hypothesis space H, which is the set of fucntions that can be learned. Moreover, most algorithms, including neural networks, learn a function $h \in H$ by formulating an optimization problem that minimizes a loss function L.

4.1 Hypothesis Space

The hypothesis space $\mathcal{H}$ is the set of all possible functions that a learning algorithm can consider as potential solutions.

Examples of hypothesis spaces:

1. Linear functions: $\mathcal{H} = \{h(x) = w_1x + w_0 | w_1, w_0 \in \mathbb{R}\}$
2. Sinusoidal functions: $\mathcal{H} = \{h(x) = A\sin(Bx + C) | A, B, C \in \mathbb{R}\}$
3. Polynomial functions: $\mathcal{H} = \{h(x) = \sum_{i=0}^{n} a_i x^i | a_i \in \mathbb{R}, n \in \mathbb{N}\}$

4.2 Loss Functions

A loss function $L(h, D)$ measures how well a hypothesis h performs on a dataset D. It quantifies the difference between predicted and actual values.

Properties of loss functions:

- Measures the discrepancy between predictions $h(\mathbf{x})$ and true labels y
- Always non-negative
- Lower values indicate better performance
- A loss of zero indicates perfect predictions

Common loss functions:

1. 0-1 Loss (for classification):

$$L_{0-1}(h, D) = \frac{1}{m}\sum_{i=1}^{m} \nVdash[h(\mathbf{x}^{(i)}) \neq y^{(i)}]$$

   where $\nVdash[\cdot]$ is the indicator function.

   This loss function is not differentiable and is typically used for evaluation rather than optimization.

2. Mean Squared Error (for regression):

$$L_{MSE}(h, D) = \frac{1}{m}\sum_{i=1}^{m}(h(\mathbf{x}^{(i)}) - y^{(i)})^2$$

   This loss function is differentiable and commonly used for regression problems.

3. Mean Absolute Error (for regression):

$$L_{MAE}(h, D) = \frac{1}{m}\sum_{i=1}^{m}|h(\mathbf{x}^{(i)}) - y^{(i)}|$$

5 Evaluating Model Performance

Proper evaluation of machine learning models is crucial to assess their generalization ability and avoid overfitting.

5.1 Train-Validation-Test Split

To evaluate model performance, the dataset D is typically divided into three subsets:

- Training set (D_{train}): Used to train the model
- Validation set (D_{val}): Used to tune hyperparameters and select the best model
- Test set (D_{test}): Used to evaluate the final model performance

These subsets should be mutually exclusive and come from the same distribution as the original dataset.

5.2 Overfitting and Underfitting

- **Underfitting:** The model has high error on both training and validation sets.
- **Overfitting:** The model has low error on the training set but high error on the validation set.
- **Good fit:** The model has low error on both training and validation sets, and generalizes well to the test set.

6 Conclusion

This introduction to machine learning covers the fundamental concepts, types of learning problems, and evaluation techniques. Understanding these basics is crucial for delving deeper into deep learning and neural networks. In the following lectures, we will explore specific algorithms, starting with linear regression, and gradually build up to more complex neural network architectures.

Exercises

1. Which of the following best describes the difference between supervised and unsupervised learning?
   (a) Supervised learning uses labeled data, while unsupervised learning uses unlabeled data
   (b) Supervised learning is always used for classification, while unsupervised learning is always used for regression
   (c) Supervised learning requires human intervention, while unsupervised learning is fully automated
   (d) Supervised learning is faster than unsupervised learning

2. In the formal definition of a supervised learning dataset $D = \{(x^{(i)}, y^{(i)})\}_{i=1}^{m}$, what does m represent?
   (a) The number of features in each example
   (b) The number of examples in the dataset
   (c) The dimensionality of the feature vector
   (d) The number of possible labels

3. Why is the 0-1 Loss function typically used for evaluation rather than optimization in classification problems?
   (a) It's too computationally expensive
   (b) It's not differentiable
   (c) It only works for binary classification
   (d) It tends to overfit the model

4. What is the primary purpose of the validation set in the train-validation-test split?
   (a) To train the model
   (b) To evaluate the final model performance
   (c) To tune hyperparameters and select the best model
   (d) To increase the overall dataset size

5. Given the hypothesis space for linear functions $H = \{h(x) = w_1 x + w_0 \mid w_1, w_0 \in \mathbb{R}\}$, which of the following correctly represents the Mean Squared Error (MSE) loss function?
   (a) $L_{MSE}(h, D) = \frac{1}{m} \sum_{i=1}^{m} |h(x^{(i)}) - y^{(i)}|$
   (b) $L_{MSE}(h, D) = \frac{1}{m} \sum_{i=1}^{m} (h(x^{(i)}) - y^{(i)})^2$
   (c) $L_{MSE}(h, D) = \frac{1}{m} \sum_{i=1}^{m} \mathbb{I}[h(x^{(i)}) \neq y^{(i)}]$
   (d) $L_{MSE}(h, D) = \frac{1}{m} \sum_{i=1}^{m} \max(0, 1 - y^{(i)} h(x^{(i)}))$