Chapter 7
Innleiðing í tilgjørdum viti

Notes

Contents

1 Introduction to AI Learning

Artificial Intelligence (AI) learning refers to the process by which an AI system improves its performance over time based on data, experience, or feedback. This process can:

- Expand the range of behaviors.
- Improve accuracy on tasks.
- Increase speed of task completion.

2 Core Components of Learning

- **Task:** The specific behavior or function being improved.
- **Data:** The experiences (examples, feedback, etc.) used to improve performance.
- **Measure of Improvement:** The metric used to quantify learning progress.

3 Large Data Requirements

AI systems often require large datasets to generalize effectively. The learner operates as a black box that takes:

- Experiences
- Problem description
- Background knowledge or bias
- And produces answers or decisions

Learner → **Models** → **Reasoner**

4 Common Learning Paradigms

- Supervised Classification
- Unsupervised Learning
- Reinforcement Learning
- Analytic Learning
- Inductive Logic Programming
- Statistical Relational Learning

5 Feedback and Bias

Feedback: Determines how success is measured.

- P: Assumes only observed negative examples are negative.
- N: Assumes only observed positive examples are positive.

Bias: Different assumptions about the distribution of examples, influencing the hypothesis space.

Example: A linear model or a polynomial fit may result from differing biases over the same data.

6 Learning as Search

Learning can be viewed as searching through a space of hypotheses or models.

- The search space is often very large.
- Techniques like gradient descent and stochastic simulation are used.

Learning algorithm = Search Space + Evaluation Function + Search Method

7 Challenges in Learning

Data Issues:

- Inadequate features
- Missing or noisy data
- Incorrectly labeled features

Types of Error:

- Limited representation (representation bias)
- Limited search (search bias)
- Limited data (variance)
- Noisy features (noise)

8 Model Representation and Bias-Variance Trade-off

- Richer representations enable better problem-solving.
- But they are harder to learn and more prone to overfitting.
- Tradeoff exists between bias (underfitting) and variance (overfitting).

9 Supervised Learning

Given a dataset of input-output pairs, the goal is to learn a function f that maps inputs to outputs.

9.1 Terminology

- Input Features
- Target Features
- Training Examples

9.2 Classification

Learning a function to map input to a discrete category (e.g., spam or not spam). Target features Y_i are discrete.

Nearest Neighbour Classification: Choose the class of the nearest training example.

K-Nearest Neighbour Classification: Choose class by majority vote among k nearest examples.

9.3 Linear Classification

A linear decision boundary:

$$x_1 = \text{humidity}, \quad x_2 = \text{pressure}$$

$$h(x_1, x_2) = \begin{cases} \text{rain} & \text{if } w_0 + w_1 x_1 + w_2 x_2 \geq 0 \\ \text{no rain} & \text{otherwise} \end{cases}$$

$$\text{Input Vector: } x = (1, x_1, x_2), \quad \text{Weight Vector: } w = (w_0, w_1, w_2)$$

$$h_w(x) = \begin{cases} 1 & \text{if } w \cdot x \geq 0 \\ 0 & \text{otherwise} \end{cases}$$

Perceptron Learning Rule:

$$w_i \leftarrow w_i + \alpha(y - h_w(x))x_i$$

9.4 Activation Functions

- **Step function:** Hard threshold.
- **Sigmoid function:** Soft threshold.

$$f(x) = \frac{1}{1 + e^{-x}}$$

Logistic Regression: Minimize the error of the logistic function by adjusting weights.

9.5 Support Vector Machines

- Adjusts the decision boundary to maximize the margin.
- **Linearly Separable:** Exists a hyperplane where classes are separable.
- Projects data into higher-dimensional space if necessary.

10 Regression

Supervised learning task to predict continuous values.

$$\text{Goal: predict target } Y \text{ from inputs } X_1, X_2, \ldots, X_n$$

Linear Regression:

$$\text{Minimize Sum of Squared Errors (SSE):} \quad SSE(E, w) = \sum_{e \in E} (o_e - p_e)^2$$

Stochastic Gradient Descent: Update weights incrementally after each example.

11 Evaluating Hypotheses

- o_e: Observed value
- p_e: Predicted value
- Error: $|o_e - p_e|$

11.1 Loss Functions

- **0-1 Loss:**
$$L(o, p) = \begin{cases} 0 & \text{if } o = p \\ 1 & \text{otherwise} \end{cases}$$
- L_1 **Loss:** Absolute error
- L_2 **Loss:** Squared error
$$L = (o - p)^2$$
- L_∞ **Loss:** Max error

12 Overfitting and Regularisation

Overfitting: Model captures noise in training data, failing to generalize.

Regularisation:

$$\text{Cost}(h) = \text{Loss}(h) + \lambda \cdot \text{Complexity}(h)$$

- Encourages simpler models.
- Feature selection: Removing irrelevant features.

13 Model Evaluation Techniques

- Holdout Validation: Train/test split.
- K-Fold Cross Validation: Split into k parts and test on each part iteratively.

14 Practical Tools

Scikit-learn: Python machine learning library offering tools for classification, regression, clustering, and evaluation.