Notes on *Mathematics For Machine Learning*

Jonathan Chen

September 12, 2025

Contents

1 Introduction and Motivation

- Machine learning designs algorithms that **automatically** extract valuable information from data. "Automatic" emphasizes general-purpose methodologies that can be applied across diverse datasets, producing meaningful outputs without heavy domain-specific customization.
- **Data**
  - ML is inherently data-driven; data forms the basis of every method.
  - Goal: uncover patterns and structure directly from datasets with minimal prior knowledge.
  - Example: topic modeling in large document corpora (Hoffman et al., 2010).
- **Model**
  - Represents the process generating the data (e.g., regression maps inputs to real-valued outputs).
  - Mitchell (1997): a model learns if its task performance improves after exposure to data.
  - Strong models must not only fit observed data but also **generalize** to unseen cases, which is essential for future applications.
- **Learning**
  - The process of optimizing model parameters to capture patterns and relationships in data.
  - Enables adaptability across tasks and datasets, reducing the need for manual rule design.
- **Mathematical Foundations**
  - Provide clarity on the principles underlying complex ML systems.
  - Enable creation of new methods beyond existing software packages.
  - Support debugging and evaluation of current approaches.
  - Reveal assumptions and limitations, which is crucial for reliable and responsible deployment in practice.

1.1 Finding words for intuition

- In machine learning, concepts and terms can be ambiguous; the same word may have different meanings depending on context.
  - Example: **algorithm**
    * As predictor: a system making predictions from input data.

    * As training procedure: a system adapting parameters so the predictor performs well on unseen data.

- The three main components of an ML system are **data**, **models**, and **learning**.
  - **Data**: represented as vectors.
    * Computer science view: array of numbers.
    * Physics view: arrow with direction and magnitude.
    * Mathematics view: object obeying addition and scaling.
  - **Model**: simplified version of the data-generating process, capturing aspects relevant for prediction and enabling exploration of hidden patterns.
  - **Learning**: training a model means optimizing its parameters with respect to a utility function measuring predictive performance.
    * Analogy: climbing a hill to maximize performance.
    * Training accuracy may only reflect memorization; the real goal is generalization to unseen data.

1.2 Two Ways to Read this Book

- **Two strategies for learning mathematics for ML**
  - Bottom-up: build from foundational to advanced concepts.
    * Advantage: solid grounding, each step relies on previous knowledge.
    * Disadvantage: foundations may feel abstract or unmotivated.
  - Top-down: start from practical needs, drill down into required math.
    * Advantage: clear motivation, direct path to applications.
    * Disadvantage: knowledge may rest on weak foundations.
- **Book structure**
  - Modular design: can be read bottom-up or top-down.
  - Part I: mathematics foundations.
  - Part II: machine learning applications (regression, dimensionality reduction, density estimation, classification).
- **Mathematical foundations (Part I)**
  - Linear algebra: vectors, matrices, data representation.
  - Analytic geometry: similarity and distance between vectors.
  - Matrix decomposition: structure and efficient computation.
  - Vector calculus: gradients for optimization.
  - Probability theory: quantifying uncertainty and noise.

  - Optimization: finding maxima/minima using gradients.

- **Applications (Part II)**
  - Regression: functions mapping inputs to outputs; MLE, MAP, Bayesian linear regression.
  - Dimensionality reduction: compact representations (e.g., PCA).
  - Density estimation: probability distributions for data (e.g., Gaussian mixtures).
  - Classification: discrete labels (e.g., support vector machines).

2 Linear Algebra

- **Algebra**: a set of objects (symbols) and rules for manipulating them.
- **Linear algebra**: study of vectors and the rules for combining them.
- **Vectors**: abstract objects that can be added and scaled (closure property). Any object satisfying these rules is a vector.
  - Geometric vectors: arrows with direction and magnitude; addition and scalar multiplication preserve vector form.
  - Polynomials: closed under addition and scalar multiplication; abstract but valid vectors.
  - Audio signals: represented as sequences of numbers; addition and scaling produce new signals.
  - Elements of $\mathbb{R}^n$: n-tuples of real numbers; focus of this book. Operations are defined component-wise.
- **Practical viewpoint**: vectors in $\mathbb{R}^n$ correspond to arrays in computer implementations. Many languages support array operations, enabling efficient ML algorithms.
- **Closure and vector spaces**: the set of all possible vectors generated by addition and scaling forms a vector space. Vector spaces and their properties underpin much of ML.
- **Role in ML**:
  - Chapter 3: analytic geometry for similarity and distances.
  - Chapter 5: matrix operations for vector calculus.
  - Chapter 9: linear regression solved via least squares.
  - Chapter 10: dimensionality reduction with projections (PCA).
  - Chapter 12: classification methods relying on linear algebra.