Mathematics and Statistics of Deep Learning

Fall 2025 (3 Credits)

1) **Instructor:** Xiongzhi Chen (xiongzhi.chen@wsu.edu)

2) **Minimal prerequisite**: Math 220 “Linear Algebra”, Math 273 “Calculus III”, Stat 360 “Probability and Statistics”

   Preferred prerequisite: Math 420 “Linear Algebra”, Math 402 “Introduction to Analysis II”, Stat 443 “Applied Probability”

   Rationale of minimal prerequisite:

   > Mathematically speaking, neural networks are constructed mainly using compositions of continuous and piece differentiable functions and matrix operations. This requires knowledge of calculus and linear algebra.

   > Computationally speaking, training of neural networks involves gradient descent or stochastic gradient descent, and their dynamics are often determined by the Hessian of a loss function. This requires knowledge of multivariate calculus, matrices as linear operators between Euclidean spaces and their fundamental properties, and convergence of infinite series.

   > Statistically speaking, neural networks learn from data that have been generated by some probability distribution, their outputs are hence random objects, and their uncertainty needs to be quantified. This requires knowledge of probability and statistics, and so does understanding and implementing stochastic gradient descent.

3) **Required textbook**: “Deep Learning: Foundations and Concepts” by Christopher M. Bishop (with Hugh Bishop), referenced as “B1” under.

4) **Recommended reference books**:
   a) “The principles of deep learning theory” by Daniel A. Roberts and Sho Yaida, referenced as “B2” under.
   b) “Deep learning” by Ian Goodfellow, Yoshua Bengio, and Aaron Courville, referenced as “B3” under.

c) “The Elements of Statistical Learning” by Trevor Hastie, Robert Tibshirani, and Jerome Friedman, references as “B4” under.

5) Course description

Deep learning, i.e., machine learning via deep neural networks, has revolutionized the art of prediction, including classification and regression, and thus has been widely deployed in biology, engineering, mathematics, statistics, chemistry, etc, whenever something is to be learnt. However, deep neural networks are mathematical and statistical models with enormous complexity, and still their inner working mechanisms have been very poorly understood theoretically, except for very shallow neural networks and despite a body of empirically found principles.

The US National Foundation has established various funding programs for the scientific community to establish a mathematical and statistical foundation for deep learning, since such a foundation will provide rigorous characterizations of the performance of these networks, help adaptively design such networks for specific tasks with little sacrifice in performance (thus leading to great saving in resources), contribute to new mathematical and statistical knowledge for the whole scientific community, and boost the productivity of any field that employs these networks.

It is very important for undergraduate and graduate students to learn the basics of how neural networks are mathematically formulated (i.e., represent neural networks as mathematical operators between two (sensible) spaces), how they are implemented (i.e., how neural networks are trained via (stochastic) gradient descent), and how the uncertainty associated with their outputs can be assessed (i.e., how to assess neural networks as statistical models in the framework of statistical learning theory). To understand these basics aspects, tools from linear algebra, calculus, probability, and statistics are needed. This leads to the minimal prerequisite Math 220 “Linear Algebra”, Math 273 “Calculus III”, Stat 360 “Probability and Statistics” for undergraduate students, and preferred prerequisite Math 420 “Linear Algebra”, Math 402 “Introduction to Analysis II”, Stat 443 “Applied Probability” for graduate students.

6) Learning outcomes

a) Students will be able to mathematically formulate a neural network as a logic sequence of mathematical operations on specific function spaces and vector spaces.

b) Students will be able to mathematically formulate the basic principles of statistical learning and how these principles are used in constructing, training, and assessing neural networks.
c) Students will be able to mathematically formulate gradient descent, stochastic gradient descent, and backpropagation for neural networks training.
d) Students will be able to use software Keras (or their preferred software) to implement training and validating neural networks via gradient descent (or stochastic gradient descent) and backpropagation
e) Students will be able to mathematically and statistically analyze shallow neural networks within the framework of statistical learning.
f) Students will be able to well understand and use convolutional neural works and graph neural networks.

7) Assessment of learning outcomes

a) In-class discussions for each lecture (discussion questions will be assigned prior to each lecture)
b) Homework assignments, covering both method, theory, and software implementation (around 4 homework assignments)
c) Midterm exam or final exam (to be determined upon discussion)
d) Presentations on course materials or research papers (to be determined upon discussion)
e) Group project (to be determined upon discussion)

8) Tentative schedule

Part I: Mathematics of neural networks

a) Single-layer networks (Chapter 4 and 5 of B1)
b) Deep neural networks (Chapter 6 of B1)
c) Gradient descent (Chapter 7 of B1)
d) Backpropagation (Chapter 8 of B1)

Part II: Statistics and software implementation of neural networks

e) Basics of statistical learning theory (Chapter 2 of B4; Section 2.5 and 2.6 of B1)
f) Statistics of single-layer networks (instructor prepared materials)

g) Empirical statistics of deep networks (instructor prepared materials)
h) Statistical theory of simplified models of networks (instructor prepared materials)
i) Sampling (Chapter 14 of B1)

Part III: Mathematics and statistics of special neural networks

j) Convolutional neural networks (Chapter 10 of B1)
k) Graph neural networks (Chapter 13 of B1)
l) Transformers (Chapter 12 of B1)
m) Generative adversarial networks (Chapter 17 of B1)

Note: We may not have enough time to cover these 4 types of neural networks. But let us try to at least cover convolutional networks and graph neural networks