ECE 510: Deep Learning Theory and Practice

Ted Willke

Spring 2019

E-mail: willke@pdx.edu
Office Hours: M/W 5:30-6:30 pm
Office: FAB-85-03

web.cecs.pdx.edu/~willke/courses/510/
Class Hours: M/W 6:40-8:30 pm
Class Room: EB102

Course Description

This course provides an introduction to the theory and practice of deep learning, with an emphasis on deep neural network-based approaches. We will start by reviewing the principles of machine learning through the lens of neural networks and deep learning. We will then move on to summarize the state of modern deep learning and familiarize ourselves with the most prominent models, such as convolutional and recurrent neural networks. We will round out the course with topics that are the subject of current research, such as representation learning and deep generative models. As a student, you can expect to learn the concepts, methods, and techniques necessary to put deep learning to work in modern applications. You will gain hands-on experience with deep learning frameworks through challenging programming projects and exercises. Lecture tutorials will provide an on-ramp for programming assignments and course projects. You will come away from this course with the ability to design, train, and evaluate deep learning models for a wide variety of applications.

Prerequisite(s): You will need familiarity with calculus (differentiation, chain rule, etc.), linear algebra, and probability. Most of the libraries and frameworks (e.g., PyTorch) are in Python. These will be used in lectures and programming assignments. You will need to be able to utilize these.

Credit Hours: 4
Required Text: There is no required textbook for this course. Lecture notes and handouts will be provided.

Primary References:


Other References:

- *Linear Algebra and Learning from Data*, Gilbert Strang, ISBN: 9780692196380
Grade Distribution:

<table>
<thead>
<tr>
<th>Component</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Assignments</td>
<td>50%</td>
</tr>
<tr>
<td>Final Project</td>
<td>30%</td>
</tr>
<tr>
<td>Quizzes</td>
<td>20%</td>
</tr>
</tbody>
</table>

**Grading Policy:** Final grade cutoffs will be 90/80/70% or lower. Project and quiz scores may be standardized.

**Lectures:**
- We will post slides after each lecture.

**Assignments**
- Homework will be assigned weekly for most weeks. The problems will be posted each Monday to the course website and due the following Monday by 11:59 AM.
- For each homework, we will post a PDF for the assignment along with starter code.

**Project**
- A final project will be submitted at the end of the course. You will research, design, and implement a deep learning model of your choice. You should analyze your model and evaluate its performance on one or more publicly-available datasets.
- You will have the opportunity to pitch your project to us early in the term and seek approval for the topic.
- You will present your project in class during the last week of the term.

**Quizzes**
- We will have several short in-class quizzes on the material presented in class and the readings. You are allowed to bring in one double-sided page of notes. There will be no midterm or final exam.

**Collaboration**
- All homework should be done individually.
- For coding assignments, students are encouraged to help one another debug, but you must write your own code. If you copy, you will receive a zero.
- For the final project, you may work in groups of up to 3 people. Each group will submit a report and give a presentation together. Our expectations for the scope and complexity of the project will increase with the number of students involved. A group of 2 may work best. We expect each student to define the role they took in the project in a paragraph to be included in the final report.
Late Policy

- Late homework assignments will not be accepted. Instead, the lowest homework score will be dropped.

Student Conduct: Students must abide by the Portland State University Code of Student Conduct, found at https://www.pdx.edu/dos/codeofconduct. Violations of this code will be handled according to procedure.

Access & Inclusion for Students with Disabilities: PSU values diversity and inclusion; we are committed to fostering mutual respect and full participation for all students. My goal is to create a learning environment that is equitable, useable, inclusive, and welcoming. If any aspects of instruction or course design result in barriers to your inclusion or learning, please notify me. The Disability Resource Center (DRC) provides reasonable accommodations for students who encounter barriers in the learning environment.

If you have, or think you may have, a disability that may affect your work in this class and feel you need accommodations, contact the Disability Resource Center to schedule an appointment and initiate a conversation about reasonable accommodations. The DRC is located in 116 Smith Memorial Student Union, 503-725-4150, drc@pdx.edu, https://www.pdx.edu/drc.

- If you already have accommodations, please contact me to make sure that I have received a faculty notification letter and discuss your accommodations.

- Students who need accommodations for tests and quizzes are expected to schedule their tests to overlap with the time the class is taking the test.

- For information about emergency preparedness, please go to the Fire and Life Safety webpage (https://www.pdx.edu/environmental-health-safety/fire-and-life-safety) for information.

Title IX Reporting Obligations As an instructor, one of my responsibilities is to help create a safe learning environment for my students and for the campus as a whole. Please be aware that as a faculty member, I have the responsibility to report any instances of sexual harassment, sexual violence and/or other forms of prohibited discrimination. If you would rather share information about sexual harassment, sexual violence or discrimination to a confidential employee who does not have this reporting responsibility, you can find a list of those individuals. For more information about Title IX please complete the required student module Creating a Safe Campus in your D2L.
## Tentative Course Outline:

<table>
<thead>
<tr>
<th>Lecture</th>
<th>Content</th>
</tr>
</thead>
</table>
| 1       | - Course overview  
         | - The learning problem  
         | - Types of learning  
         | - Tutorial: Perceptron learning algorithm  
| 2       | - Generalization  
         | - Linear regression and gradient descent  
         | - Tutorial: Linear regression with perceptron  
| 3       | - Error metrics  
         | - capacity, underfitting, and overfitting  
         | - Tutorial: Multilayer Perceptron  
| 4       | - Backpropagation  
         | - Stochastic gradient descent  
         | - Regression for classification  
         | - Tutorial: Neural network classifier  
| 5       | - Theory of generalization  
         | - Approximation capability of neural networks  
         | - Generalization and the VC-dimension  
         | - Tutorial: Assessing performance on image data  
| 6       | - Regularization (weight decay and early stopping)  
         | - Hyperparameters and validation sets  
         | - Estimators, bias, and variance  
         | - Tutorial: Cross-validation of a neural network  
| 7       | - Challenges in neural network optimization  
         | - Learning rates, momentum, and adaptive learning  
         | - Initialization strategies and batch normalization  
         | - Tutorial: Evaluating optimization strategies  
| 8       | - Reducing generalization error  
         | - Parameter norm penalties  
         | - Data augmentation, noise injection, and dropout  
         | - Tutorial: Regularizing neural networks  
| 9       | - Convolutional neural networks (CNNs)  
         | - Parameter tying and sharing  
         | - The convolution operation and its motivation  
         | - Pooling and convolutional layer models  
         | - Tutorial: Classifying images with a CNN  
| 10      | - Variants of basic convolution  
         | - Data types (1-D, 2-D, 3-D)  
         | - neural basis for convolution  
<pre><code>     | - Tutorial: Signal processing with a CNN |
</code></pre>
<table>
<thead>
<tr>
<th>Lecture</th>
<th>Content</th>
</tr>
</thead>
</table>
| 11 | • Recurrent neural networks (RNNs)  
• Unfolding computation graphs  
• Computing the gradient in RNNs  
• Bidirectional RNNs  
• Tutorial: Generating sequences with RNNs |
| 12 | • Project pitch day |
| 13 | • Topics in deep RNNs  
• Recursive neural networks  
• Vanishing and exploding gradients  
• Gated RNNs  
• Tutorial: Language processing with LSTMs |
| 14 | • Long-term dependencies and explicit memory  
• Neural basis for complementary memory  
• Linear factor models  
• Tutorial: Hyperparameter tuning, performance metrics, and debug |
| 15 | • Autoencoders (AEs) and dimensionality reduction  
• Undercomplete, sparse, and regularized autoencoders  
• Variational autoencoders  
• Tutorial: AE dimensionality reduction and t-SNE |
| 16 | • Deep generative models  
• Boltzmann machines, restricted Boltzmann machines  
• Deep belief networks and deep Boltzmann machines  
• Tutorial: Making predictions with RBMs |
| 17 | • Generative adversarial networks (GANs)  
• Representation learning  
• Transfer learning |
| 18 | • Reinforcement learning (RL)  
• Markov decision processes  
• Tutorial: Solving problems with RL agents |
| 19 | • Deep reinforcement learning  
• Temporal-difference learning  
• Deep Q-learning  
• Tutorial: Playing games with DQNs |
| 20 | • Final project presentations |