

**2024 Winter Semester: the course will be given in English**

**Course Name: Deep Learning**

**English syllabus:**

We will learn theoretical and practical tools to build, design and analyze deep networks, with an emphasis on supervised learning. For example, properties and convergence of gradient descent and its variants, efficient differentiation, multilayer nets (approximation and symmetry), convnets (and extensions) for visual tasks, training methods and their analysis, networks for serial data, and pre-training.

Subject Outline for the Course:

1. Introduction - 2 hours .Review of learning systems, inclusion, the importance of deep learning, and course overview.
2. Optimization - 2 hours. Properties and promises of various optimization algorithms: gradient descent, stochastic gradient descent, momentum, adaptive learning rates, etc.
3. Efficient Differentiation - 2 hours. Backpropagation using chain rule, differentiation in general graphs and implicit functions, sub-gradients, numerical differentiation, second-order derivatives.
4. Single Neuron - 2 hours. Regression and classification cost functions, activation functions, optimization surfaces, convergence, hidden regularization, and inclusion.
5. Multilayer Neural Networks - 2 hours. Universal approximation theorem and its extensions, optimization surfaces, symmetry and preserved dimensions, initialization methods, skip-connections.
6. Networks for Vision Tasks - 2 hours. Convolutional networks, detection, segmentation; augmentation methods.
7. Networks for Sequential Tasks - 2 hours. Feedback networks, attention mechanisms, transformers.
8. Training Methods - 2 hours. Input normalization, normalization layers, regularization, optimization methods, various cost functions, hyperparameter tuning.
9. Impact Analysis of Training Methods - 2 hours. Interaction between factors, analysis of simple networks.
10. Data Efficiency and Pre-training - 2 hours. Self-supervision, representation learning, transfer learning.
11. ~~Resource Efficiency and Model Compression - 2 hours. —Model pruning, dilution, quantization (not part of Winter 2024 semester material).~~
12. Summary Lecture, Selected Topics, Project Presentations - 2 hours.
13. Project Presentations - 3 hours.

**Learning Outcomes:**

With the completion of the course, the students:

1. Will be familiar with the main models and common training methods for deep learning.
2. Will be able to code (in Python, using the Pytorch framework) for deep neural network, train it, and use it.
3. Will be able to understand the considerations required to tune deep networks for achieving good performance, and the relevant theoretical results (when such exist).

**External Resources:**

[Dive into Deep Learning](#), Aston Zhang, Zack C. Lipton, Mu Li, Alex Smola, 2020

[Machine Learning with PyTorch and Scikit-Learn](#), Sebastian Raschka, *Packt*, 2022

[Understand Deep Learning](#), Simon J.D. Prince, MIT Press, 2023

[Deep Learning](#), Ian Goodfellow, Yoshua Bengio, and Aaron Courville, *MIT Press*, 2016

### **More Information:**

**Staff:** lectures will be given by Prof. Daniel Soudry, tutorials will be given by Tal Daniel (Head TA) and the HW will be graded by Noam Kasten. Details on contacting us are available on the Moodle.

**Registration and Waiting List:** in its current form, the course can support 40 enrolled students. There is a registration waiting list of students that were not able to register but want to join should students cancel their registration. **We ask you to make the decision to stay or leave as soon as possible and notify the TA, Tal, via email if you decide to leave the course in order to allow students in the waiting list to join the course.**

**Winter 2024 Adjustments:** the course will be given in **English** (both in-class and recordings). The semester is shortened to 12 weeks, the subject of the final week (“Compression, pruning, quantization”) is off the syllabus, but we will provide the material for completeness for the curious students.

**In-Class Teaching:** as much as we want to have you all in-class for the lectures/tutorials (teaching in front of black screens is not that fun), we understand that some of you prefer/learn better at home. We will post recordings of the lectures and tutorials (some may be from the previous semester), and we will also try to use recordings for parts that require more time or were less understood in class (mainly for the tutorials) Recordings will be posted after Daniel’s lecture. Schedule:

- **Lectures** –Daniel Soudry - Tuesdays, 10:30-12:30, in Meyer 352.
- **Tutorials** –Tal Daniel - Tuesdays, 12:30-13:30, in Meyer 352.

**Recorded Material:** Recorded material will be published after the in-class sessions. **Important note:** the course material is updated every semester as the deep learning field is moving very fast. As such, the recordings might not include all of the subjects covered in-class and in the written material and required for the HW and exam. Students that solely rely on recordings should cover the missing topics on their own as they are part of the syllabus.

### **Grade Structure:**

- HW – 30% (HW0 binary, HW1-3 10% each, must pass the final exam)
- Project – 30% (must pass the final exam)
- Final Exam – 40%
- Preparation/Practice exercised on Moodle – 2% **Bonus** (must pass the final exam)
- **Note: the final grade of students that fail either the final exam or the final project will be the lower between the two (no HW/project/bonus consideration).**

- **Winter 2024: students can choose not to submit HW3, changing the HW weight to 20% (HW 1-2) and the exam to 50%.**

**Working Environment:** the lectures will be in a PowerPoint slides format and will be published on the Moodle website. The tutorials will be in a Jupyter Notebook format, including both math and Python code. PDF version of the tutorials will be uploaded to the Moodle, and the raw notebooks will be available on the course's GitHub (see below). For HW, you will also be required to write in Python and submit Jupyter Notebooks (you can use any IDE you like, but the final submission must be in a Jupyter Notebook). We have prepared a manual (with video tutorial) for you how to set-up a working environment on your local machine or work remotely on Google Colab (which will also give you a GPU), and you can find it on Moodle.

**HW:** there will be 4 HW exercises. HW0 is a binary-graded exercise to make sure you know what Python is. HW1-HW3, each 10% of your grade, will include both theoretical exercises ("dry") to practice what was learned in the lectures and hand-on computer assignments ("wet") to hone your deep learning skills, which will also be practiced in the tutorials (**Winter 2024: students can choose not to submit HW3, changing the HW weight to 20% (HW 1-2) and the exam to 50%**). **Submission is only in pairs.** You can change pairs between assignments. For each assignment, you will need to create a group in the Moodle and then submit through this group (so only one of the pair can submit and both of you will get the grade). The HW will be provided in a Jupyter Notebook format, however, for the theoretical part, you can submit a separate PDF with your answers (you can use Word, LyX, Latex, Overleaf and etc..., **but no hand-written submissions!**).

**Project:** when you finish this course, we want you to have "proof" that you are qualified to work on deep learning projects. In our case, this will be in the form of a practical project. This is important for your CV, job interviews and etc. We will have a "project orientation" in the middle of the semester, and this will be the point where you should start working on it. We will publish project proposals, but you can suggest your own. You will submit your proposals around week 6-7, and we will make sure the scale of the projects is valid for the timeframe of the course. We will also hold personal meetings with the students (pairs only) to help them. In the final week of the semester, the students will present their projects, and submit it by the end of the semester.

**Preparation/Practice Exercises on Moodle:** we will also have non-mandatory Moodle exercises to practice or prepare for the class. These will be short questions on different topics (according to weeks), some of them are "job interview" level. For completing them, you can earn up to 2% Bonus for the final grade.

**Forum:** our forum will be hosted on Piazza, please register with your @campus emails. If you are new to Piazza, please see our short video tutorial on Moodle.

**Office Hours:** we will be happy to set-up office hours with you. Please set them up with us via emails. Moreover, for any personal problem/concern you have, please feel free to contact me (Tal). For other matters, please consult with your semester representative and they will contact us.

**Reserve Duty (Miluim):** please reach out to us privately if you are on active duty and require personal adjustments, we will do our best to help you.

**שם עברי מקוצר:** למידה עמוקה

### **דרישות קדם:**

"מערכות לומדות" 046195, או "מבוא למערכות לומדות" 236756 בקורס זה נתכנת ב-Python ונשתמש בסביבת Pytorch, כפי שנלמד בקורס מערכות לומדות (מאז אביב 2019). חומר עזר יינתן לסטודנטים שצריכים רענון, אך באחריות הסטודנטים להשלים כל חוסרים.

**סמסטר חורף 2024: הקורס יועבר באנגלית**

### **סילבוס בעברית**

נלמד כלים תיאורטיים ומעשיים, כדי לבנות, לאמן ולנתח רשתות נוירונים ללמידה עמוקה, בדגש על למידה מונחית. למשל, תכונות ותנאי התכנסות של אלגוריתם הגרדיאנט וגרסאותיו השונות, רשתות רב שכבתיות (תכונות קירוב וסימטריות), חישוב יעיל של נגזרות, רשתות קונוולוציה והרחבותיהן למשימות ראייה, שיטות אימון וניתוחן, רשתות למשימות סדרתיות, ולמידה מקדימה.

### **פרט נושאי הלימוד והערכת משכם:**

1. הקדמה – 2 שעות.  
חזרה על מערכות לומדות, הכללה, מדוע למידה עמוקה חשובה, מה נעשה בקורס.
2. אופטימיזציה – 2 שעות.  
תכונות והבטחות התכנסות של אלגוריתמי אופטימיזציה שונים: אלגוריתם הגרדיאנט, גרדיאנט סטוכסטי, מומנטום, קצב לימוד מסתגל, וכו'.
3. גזירה יעילה – 2 שעות.  
פעפוע אחורי (Backpropagation) באמצעות כופלי לגרנז', גזירה בגרפים כלליים ופונקציות סתומות, תת-גרדיאנטים, גזירה נומרית, נגזרות מסדר שני.
4. נוירון בודד – 2 שעות.  
פונקציות מחיר לרגרסיה וסיווג, פונקציות אקטיבציה, משטח האופטימיזציה, התכנסות, רגולריזציה חבויה, הכללה.
5. רשתות נוירונים רב שכבתיות – 2 שעות.  
משפט הקירוב האוניברסלי והרחבותיו, משטח האופטימיזציה, סימטריות וגדלים נשמרים, שיטות אתחול, חיבורי דילוג (skip-connections).
6. רשתות למשימות ראייה – 2 שעות.  
רשתות קונבולוציה, זיהוי, וסגמנטציה; שיטות אוגמנטציה
7. רשתות למשימות סדרתיות – 2 שעות.  
רשתות עם משוב, קשב (attention), ושנאים (transformers)
8. שיטות אימון – 2 שעות.  
נרמול הקלט, שכבות נרמול, רגולריזציה, שיטות אופטימיזציה, פונקציות מחיר שונות, וכיוונון היפר-פרמטרים.
9. ניתוח השפעה של שיטות אימון – 2 שעות.  
אינטראקציה בין גורמים, ניתוח של רשתות פשוטות.
10. יעילות בנתונים ולמידה מקדימה (pre-training) – 2 שעות.  
הנחיה עצמית, למידת ייצוג, למידת העברה (transfer learning)
11. **יעילות משאבים ודחיסת מודלים – 2 שעות**  
**דיקוק מודלים, דילוג, קוונטיזציה (ירד מהחומר של סמסטר חורף 2024)**
12. הרצאת סיכום, נושאים נבחרים, הצגת פרויקטים – 2 שעות
13. הצגת פרויקטים – 3 שעות.

### מקורות:

הקורס אינו עוקב אחר מקור מסויים, אולם המקורות הבאים יכולים להיות שימושיים:

[Dive into Deep Learning](#), Aston Zhang, Zack C. Lipton, Mu Li, Alex Smola, 2020

[Machine Learning with PyTorch and Scikit-Learn](#), Sebastian Raschka, *Packt*, 2022

[Understand Deep Learning](#), Simon J.D. Prince, MIT Press, 2023

[Deep Learning](#), Ian Goodfellow, Yoshua Bengio, and Aaron Courville, *MIT Press*, 2016

### תוצאות למידה:

עם השלמת הקורס בהצלחה, הסטודנטים יהיו מסוגלים:

1. להכיר את המודלים העיקריים ושיטות אימון מקובלות ללמידה עמוקה.
2. לכתוב קוד python, בסביבת pytorch, לבנייה, אימון ושימוש ברשת עמוקה.
3. להבין את השיקולים הנדרשים לכיווןן רשתות עמוקות כדי לקבל ביצועים טובים, ואת התוצאות התיאורטיות הרלוונטיות (במידה וקיימות).

### הרכב הציון:

**במידה ואין ציון נכשל בבחינה הסופית או בפרויקט, הציון יקבע לפי:**

40% בחינה סופית

30% שיעורי בית הכוללים חלק "יבש" ו"רטוב" (ש"ב 0 בינארי "חייב לעבור", ש"ב 1-3 10% כל אחד)

30% פרויקט סופי

2% בונוס שאלות תרגול הכנה במודל

**במידה ויש ציון נכשל בבחינה הסופית או בפרויקט הסופי – הציון הסופי יהיה הנמוך מבין השניים.**