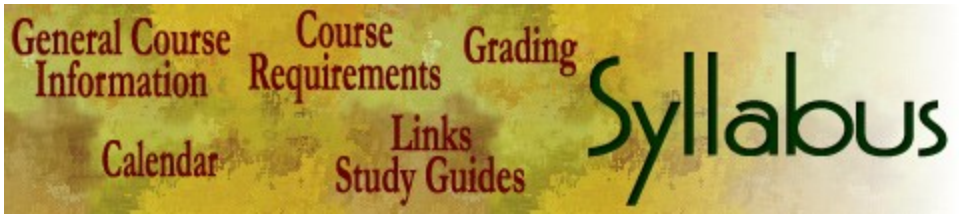

University of Florida

Department of Electrical and Computer Engineering

EEL 6825, Section 026A

Pattern Recognition and Intelligent Systems

Spring 2020

Course Description

This is a 3-credit course.

The objective of this course is to impart a working knowledge of several important and widely used pattern recognition topics to the students through a mixture of motivational applications and theory.

Course Prerequisites

- EEL 3135 (Discrete-Time Signals and Systems) or undergraduate-level signals and systems
- EEL 4516 (Noise in Devices and communication Systems) or undergraduate-level probability theory/stochastic processes
- Some exposure to MATLAB and C programming language
- Knowledge of basic matrix theory (linear algebra) would be helpful, but not necessary

Required Textbook

- Richard O. Duda, Peter E. Hart, David G. Stork, ``Pattern Classification,'' 2nd Edition, Wiley-Interscience, October 2000. ISBN-10: 0471056693 | ISBN-13: 978-0471056690.

Recommended Readings

- David G. Stork, Elad Yom-Tov, ``Computer Manual in MATLAB to accompany Pattern Classification,'' 2nd Edition, Wiley-Interscience, April 2004. ISBN: 978-0-471-42977-7
- Christopher M. Bishop, “Pattern Recognition and Machine Learning”, 1st Edition, Springer, October 1, 2007. ISBN-10: 0387310738 | ISBN-13: 978-0387310732.

- Trevor Hastie, Robert Tibshirani, Jerome Friedman, "The Elements of Statistical Learning: Data Mining, Inference, and Prediction," Second Edition, Springer, February 9, 2009. ISBN-10: 0387848576 | ISBN-13: 978-0387848570.
- Kevin Patrick Murphy, "Machine Learning: a Probabilistic Perspective," the MIT Press, August 24, 2012. ISBN-10: 0262018020 | ISBN-13: 978-0262018029.
- Eugene Charniak, "Introduction to Deep Learning," the MIT Press, January 2019. ISBN: 9780262039512 (This book is a project-based guide to the basics of deep learning)

Course Information

Instructor:

Dr. Dapeng Wu
Office: NEB 431
Email: wu@ece.ufl.edu

TA:

1) Heng Qiao
Email: hengqiao@ufl.edu

2) Haotian Jiang
Email: haotian.jiang@ufl.edu

3) Xiyao Ma
Email: maxiy@ufl.edu

4) Tong Shao
Email: stlm1991@ufl.edu

Course website: http://www.wu.ece.ufl.edu/courses/eel6825s20

Meeting Time

Monday, Wednesday, Friday, period 8 (3:00 pm - 3:50 pm)

Meeting Room

NEB 100

Office Hours

- Dr. Wu: Monday, Wednesday, period 9 (4:05 pm - 4:55 pm), and by appointment via email.

Structure of the Course

The course consists of lectures, 4 homework assignments, and 1 project.

This course is primarily a lecture course. I cover all important material in lectures. Since EEL 3135 and EEL 4516 are prerequisites, I assume some previous knowledge about DSP, probability theory and stochastic processes, and hence I will cover some material very quickly. Thus, depending on what and how much you recall from earlier study, varying amounts of reading in introductory books on DSP, probability theory and stochastic processes (other than the course textbook) may be necessary; these readings are up to the student. I will only give reading assignments from the course textbook.

Attending lecture is quite important as I may cover material not available in any book easily accessible to you. I use Powerpoint presentation during lecture. Lecture notes will be posted on the course website before the class. The lecture is to engage the students in independent thinking, critical thinking, and creative thinking, help the students organize the knowledge around essential concepts and fundamental principles, and develop conditionalized knowledge which tells them when, where and why a certain method is applicable to solving the problem they encounter.

I do not intend for the WWW material to be a substitute for attending lecture since engaging the students in active thinking, making logical connections between the old knowledge and the new knowledge, and providing insights are the objectives of my lecture. The lecture notes are posted on the web so that you can miss an occasional lecture and still catch up, and it makes taking notes easier. To reward those who attend regularly, there will be some lecture-based material in the exam which is not available via the web.

The class project is described here.

Course Outline

- Bayesian decision theory
- Parametric estimation and supervised learning
- Nonparametric methods
- Linear discriminant functions
- Unsupervised learning and clustering
- Nonmetric methods
- Feature extraction and feature selection
- Applications

Course Objectives

Upon the completion of the course, the student should be able to

- use the fundamental techniques for pattern recognition
- understand the basics of statistical learning theory
- acquire the basic skill of designing machine learning algorithms and systems

Handouts

Please find handouts here.

Requirements

Course Policies

- **Attendance:**
  - Perfect class attendance is not required, but regular attendance is expected.
  - It is the student's responsibility to independently obtain any missed material (including handouts) from lecture.
- **During lecture, cell phones should be turned off.**
- **No late submissions of your homework solution, and project proposal/report, are allowed unless U.F. approved reasons are supplied and advance permission is granted by the instructor. Excused late submissions are consistent with university policies in the undergraduate catalog (https://catalog.ufl.edu/ugrad/current/regulations/info/attendance.aspx) and require appropriate documentation.**
- **Software use**
  - All faculty, staff and student of the University are required and expected to obey the laws and legal agreements governing software use. Failure to do so can lead to monetary damages and/or criminal penalties for the individual violator. Because such violations are also against University policies and rules, disciplinary action will be taken as appropriate. We, the members of the University of Florida community, pledge to uphold ourselves and our peers to the highest standards of honesty and integrity.
- **Announcements:**
  - All students are responsible for announcements made in lecture, on the student access website, or via the class email list.
  - It is expected that you will check your email several times per week for possible course announcements.
- **Students with disabilities:**
  - Students with disabilities requesting accommodations should first register with the Disability Resource Center (352-392-8565, https://www.dso.ufl.edu/drc) by providing appropriate documentation. Once registered, students will receive an accommodation letter which must be presented to the instructor when requesting accommodation. Students with disabilities should follow this procedure as early as possible in the semester.

- **University Honesty Policy**

  UF students are bound by The Honor Pledge which states, “We, the members of the University of Florida community, pledge to hold ourselves and our peers to the highest standards of honor and integrity by abiding by the Honor Code. On all work submitted for credit by students at the University of Florida, the following pledge is either required or implied: “On my honor, I have neither given nor received unauthorized aid in doing this assignment.” The Honor Code (https://www.dso.ufl.edu/sccr/process/student-conduct-honor-code/) specifies a number of behaviors that are in violation of this code and the possible sanctions. Furthermore, you are

obligated to report any condition that facilitates academic misconduct to appropriate personnel. If you have any questions or concerns, please consult with the instructor or TAs in this class.

Students are encouraged to discuss class material in order to better understand concepts. All homework answers must be the author's own work. However, students are encouraged to discuss homework to promote better understanding. What this means in practice is that students are welcome to discuss problems and solution approaches, and in fact can communally work solutions at a board. However, the material handed in must be prepared starting with a clean sheet of paper (and the author's recollection of any solution session), but not refer to any written notes or existing code from other students during the writing of the solution. In other words, writing the homework report shall be an exercise in demonstrating the student understands the materials on his/her own, whether or not help was provided in attaining that understanding.

All work submitted in this course must be your own and produced exclusively for this course. The use of sources (ideas, quotations, paraphrases) must be properly acknowledged and documented. For the copy of the UF Honor Code and consequences of academic dishonesty, please refer to http://www.dso.ufl.edu/sccr/honorcodes/honorcode.php. Violations will be taken seriously and are noted on student disciplinary records. If you are in doubt regarding the requirements, please consult with the instructor before you complete any requirement of the course.

Course Evaluation

Students are expected to provide feedback on the quality of instruction in this course by completing online evaluations at https://evaluations.ufl.edu/evals. Evaluations are typically open during the last two or three weeks of the semester, but students will be given specific times when they are open. Summary results of these assessments are available to students at https://evaluations.ufl.edu/results/.

Software Use

All faculty, staff, and students of the University are required and expected to obey the laws and legal agreements governing software use. Failure to do so can lead to monetary damages and/or criminal penalties for the individual violator. Because such violations are also against University policies and rules, disciplinary action will be taken as appropriate. We, the members of the University of Florida community, pledge to uphold ourselves and our peers to the highest standards of honesty and integrity.

Student Privacy

There are federal laws protecting your privacy with regards to grades earned in courses and on individual assignments. For more information, please see:
http://registrar.ufl.edu/catalog0910/policies/regulationferpa.html

Campus Resources:

Health and Wellness

U Matter, We Care:

If you or a friend is in distress, please contact umatter@ufl.edu or 352 392-1575 so that a team member can reach out to the student.

Counseling and Wellness Center: http://www.counseling.ufl.edu/cwc, and 392-1575; and the University Police Department: 392-1111 or 9-1-1 for emergencies.

Sexual Assault Recovery Services (SARS)

Student Health Care Center, 392-1161.

University Police Department at 392-1111 (or 9-1-1 for emergencies), or http://www.police.ufl.edu/.

Academic Resources

E-learning technical support, 352-392-4357 (select option 2) or e-mail to Learning-support@ufl.edu. https://lss.at.ufl.edu/help.shtml.

Career Resource Center, Reitz Union, 392-1601. Career assistance and counseling. https://www.crc.ufl.edu/.

Library Support, http://cms.uflib.ufl.edu/ask. Various ways to receive assistance with respect to using the libraries or finding resources.

Teaching Center, Broward Hall, 392-2010 or 392-6420. General study skills and tutoring. https://teachingcenter.ufl.edu/.

Writing Studio, 302 Tigert Hall, 846-1138. Help brainstorming, formatting, and writing papers. https://writing.ufl.edu/writing-studio/.

Student Complaints Campus: https://www.dso.ufl.edu/documents/UF_Complaints_policy.pdf.

On-Line Students Complaints: http://www.distance.ufl.edu/student-complaint-process.

TOP

Grading

Grading:

Grades	Percentage	Dates
Homework	30%	see calendar

Project proposal	10%	4pm, March 13
Project report	60%	4pm, April 29

The project report consists of

1. (50%) A written report for your project
2. (25%) Computer programs that you develop for your project
3. (10%) Powerpoint file of your presentation
4. (15%) Your presentation/demo video on YouTube

Grading scale:

Top 25% students will receive A. Average score will be at least B+.

More information on UF grading policy may be found at:
https://catalog.ufl.edu/ugrad/current/regulations/info/grades.aspx

Homework:

- Due dates of assignments are specified in the course calendar.
- **No late submissions** are allowed unless U.F. approved reasons are supplied and advance permission is granted by the instructor.
- If you wish to dispute a homework grade, you must return the assignment along with a succinct written argument within one week after the graded materials have been returned to the class. Simple arithmetic errors in adding up grade totals are an exception, and can normally be handled verbally on-the-spot during office hours of the TA. For all other disputes, the entire homework may be (non-maliciously) re-graded, which may result in increase or decrease of points.

Class Project:

The class project will be done individually. Each project requires a proposal and a final report. The final report is expected to be in the format of a conference paper plus computer programs, a Powerpoint file, and a video. On March 13, the project proposal (up to 2 pages) is due. On April 29, the final report (up to 10 pages) is due. For details about the project, please read here.

Suggested topics for projects are listed here.

TOP

Calendar

Course calendar can be found here.

Links Study Guides

Useful links

- Anaconda: Anaconda is the leading open data science platform powered by Python.
- Theano: Theano is a Python library that lets you to define, optimize, and evaluate mathematical expressions, especially ones with multi-dimensional arrays (numpy.ndarray).
- TensorFlow: TensorFlow is an open source software library for numerical computation using data flow graphs. Nodes in the graph represent mathematical operations, while the graph edges represent the multidimensional data arrays (tensors) communicated between them. The flexible architecture allows you to deploy computation to one or more CPUs or GPUs in a desktop, server, or mobile device with a single API.
- Keras: Keras is a minimalist, highly modular neural networks library, written in Python and capable of running on top of either TensorFlow or Theano. It was developed with a focus on enabling fast experimentation. Being able to go from idea to result with the least possible delay is key to doing good research.
- PyTorch: PyTorch is a deep learning framework for fast, flexible experimentation.
- A curated list of resources dedicated to recurrent neural networks
- Source code in Python for handwritten digit recognition, using deep neural networks
- Source code in PyTorch for handwritten digit recognition, using deep neural networks
- Source code in Python for TF-mRNN: a TensorFlow library for image captioning
- Source code in Python for the following work on image captioning:
    - Oriol Vinyals, Alexander Toshev, Samy Bengio, Dumitru Erhan, Show and Tell: A Neural Image Caption Generator, CVPR 2015
        - Implementation
- Image captioning:
    - Zhe Gan, et. al, Semantic Compositional Networks for Visual Captioning, CVPR 2017
        - Implementation Source code in Python (Theano)
    - Bottom-Up and Top-Down Attention for Image Captioning and Visual Question Answering: source codes (Caffe) and source codes (PyTorch)
- Microsoft COCO datasets
- Visual Question Answering:

    - Bottom-Up and Top-Down Attention for Image Captioning and Visual Question Answering: source codes (Caffe) and VQA source code (PyTorch)

- Semantic Propositional Image Caption Evaluation (SPICE)
    - Source code in JAVA to calculate SPICE
- Region-based Convolutional Neural Networks (R-CNN)
    - References:
        - Ren, Shaoqing, Kaiming He, Ross Girshick, and Jian Sun. "Faster R-CNN: Towards real-time object detection with region proposal networks." In Advances in neural information processing systems, pp. 91-99. 2015. [pdf]

        - Dai, Jifeng, Yi Li, Kaiming He, and Jian Sun. "R-FCN: Object detection via region-based fully convolutional networks." In Advances in neural information processing systems, pp. 379-387. 2016. [pdf] [source code]
        - Huang, Jonathan, Vivek Rathod, Chen Sun, Menglong Zhu, Anoop Korattikara, Alireza Fathi, Ian Fischer et al. "Speed/accuracy trade-offs for modern convolutional object detectors." arXiv preprint arXiv:1611.10012 (2016). [pdf] (E.g., for Inception V3, extract features from the "Mixed 6e" layer whose stride size is 16 pixels. Feature maps are cropped and resized to 17x17.)
    - Source codes:
        - A Faster Pytorch Implementation of Faster R-CNN (PyTorch)
        - Bottom-Up and Top-Down Attention for Image Captioning and Visual Question Answering: source codes (Caffe)
- Source code in Python for end-to-end training of LSTM
    - Implementation
- Bidirectional Encoder Representations from Transformers (BERT)
    - Implementation in TensorFlow
    - Implementation in PyTorch
- Source code in Python for sequence-to-sequence learning (language translation, chatbot)
    - TensorFlow seq2seq library
    - Implementation 1 on Tensorflow with separable encoder and decoder
    - Implementation 2 on Keras
- Visual Storytelling Dataset (VIST)
    - Visual storytelling algorithms:
        - No Metrics Are Perfect: Adversarial REward Learning for Visual Storytelling: source codes (TensorFlow)
- Visual Genome is a dataset, a knowledge base, an ongoing effort to connect structured image concepts to language.
- MPII Movie & Description dataset for automatic video description, video summary, video storytelling
- Bidirectional recurrent neural networks (B-RNN):
    - Graves, Alan, Navdeep Jaitly, and Abdel-rahman Mohamed. "Hybrid speech recognition with deep bidirectional LSTM." IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU), 2013. [pdf]
- Deep reinforcement learning
    - References:
        - Mnih, Volodymyr, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra, and Martin Riedmiller. "Playing atari with deep reinforcement learning." *arXiv preprint arXiv:1312.5602* (2013).
        - Mnih, Volodymyr, Koray Kavukcuoglu, David Silver, Andrei A. Rusu, Joel Veness, Marc G. Bellemare, Alex Graves et al. "Human-level control through deep reinforcement learning." *Nature* 518, no. 7540 (2015): 529-533. [source code]
        - How to Study Reinforcement Learning
    - Source codes:
        - Implementation of Reinforcement Learning Algorithms. Python, OpenAI Gym, Tensorflow. Exercises and Solutions to accompany Sutton's Book and David Silver's course. [link]
- Generative Adversarial Network (GAN)
    - References:

        - Goodfellow, Ian, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. "Generative adversarial nets." In *Advances in neural information processing systems*, pp. 2672-2680. 2014.
        - Radford, Alec, Luke Metz, and Soumith Chintala. "Unsupervised representation learning with deep convolutional generative adversarial networks." *arXiv preprint arXiv:1511.06434* (2015).
        - Arjovsky, Martin, Soumith Chintala, and Léon Bottou. "Wasserstein GAN." *arXiv preprint arXiv:1701.07875* (2017).
    - Types of GAN
        1. Vanilla GAN
        2. Conditional GAN
        3. InfoGAN
        4. Wasserstein GAN
        5. Mode Regularized GAN
        6. Coupled GAN
        7. Auxiliary Classifier GAN
        8. Least Squares GAN
        9. Boundary Seeking GAN
        10. Energy Based GAN
        11. f-GAN
        12. Generative Adversarial Parallelization
        13. DiscoGAN
        14. Adversarial Feature Learning & Adversarially Learned Inference
        15. Boundary Equilibrium GAN
        16. Improved Training for Wasserstein GAN
        17. DualGAN
        18. MAGAN: Margin Adaptation for GAN
        19. Softmax GAN
    - Source codes:
        - A Tensorflow Implementation of "Deep Convolutional Generative Adversarial Networks": python code
        - Collection of generative models, e.g. GAN, VAE in Pytorch and Tensorflow: python code
- Sequential Generative Adversarial Network (GAN)
    - References:
        - Yu, Lantao, Weinan Zhang, Jun Wang, and Yong Yu. "SeqGAN: Sequence Generative Adversarial Nets with Policy Gradient." In *AAAI*, pp. 2852-2858. 2017.
        - Mogren, Olof. "C-RNN-GAN: Continuous recurrent neural networks with adversarial training." *arXiv preprint arXiv:1611.09904* (2016).
        - Im, Daniel Jiwoong, Chris Dongjoo Kim, Hui Jiang, and Roland Memisevic. "Generating images with recurrent adversarial networks." *arXiv preprint arXiv:1602.05110* (2016).
        - Press, Ofir, Amir Bar, Ben Bogin, Jonathan Berant, and Lior Wolf. "Language Generation with Recurrent Generative Adversarial Networks without Pre-training." *arXiv preprint arXiv:1706.01399* (2017).
    - Source codes:
        - Implementation of C-RNN-GAN

        - Tensorflow Implementation of GAN modeling for sequential data
- Stanford NLP Parser: A natural language parser is a program that works out the grammatical structure of sentences.
- Performance metrics for a natural language parser
- Precision and recall
- mAP (mean Average Precision) for Object Detection
- Question answering
    - References:
        - Seo, Minjoon, Aniruddha Kembhavi, Ali Farhadi, and Hannaneh Hajishirzi. "Bidirectional attention flow for machine comprehension." *arXiv preprint arXiv:1611.01603* (2016).
    - Source codes:
        - Bi-Directional Attention Flow (BIDAF)
    - Question answering datasets:
        - Stanford Question Answering Dataset (SQuAD)
        - NewsQA
        - MS MARCO
- The General Language Understanding Evaluation (GLUE) benchmark is a collection of resources for training, evaluating, and analyzing natural language understanding systems.
- Semantic Textual Similarity (STS) benchmark evaluation dataset
- Automatic text understanding and reasoning:
    - Facebook bAbI project
    - Facebook dataset
    - Python code
- NLTK sentiment analysis tool
- Opinion Lexicon (dictionary of sentiment words): Positive and Negative
- Human activity recognition
    - HMDB: a large human motion database
        - Action recognition algorithms
    - UCF101: Action Recognition Data Set
        - Activity recognition algorithms
- Batch Normalization and Weight Decay Notes
- **MATLAB Tutorial**
- **MATLAB Central**
- **Matlab Primer**, **Matlab Manuals**, **Image Processing Toolbox**
- **Matlab implementation of image/video compression algorithms**
- **Introduction to Matarix Algebra (free book by Autar K Kaw, Professor, University of South Florida).**
- **Matrix Reference Manual**
- **HIPR2: a WWW-based Image Processing Teaching Materials with J**
- **Learning by simulations**
- **OpenCV**
- **OpenGL**
- **A Recipe for Training Neural Networks (by Andrej Karpathy)**
- Download the following free (open source) program to record video with screen capture: http://www.nchsoftware.com/capture/index.html?gclid=CNadwsW6-6wCFSVjTAodbjzTSg

---

Software:

- Virtual Dub: VirtualDub is a video capture/processing utility for 32-bit Windows platforms (95/98/ME/NT4/2000/XP), licensed under the GNU General Public License (GPL).
- XnView: is an efficient multimedia viewer, browser and converter.
- ImageJ: Read and write GIF, JPEG, and ASCII. Read BMP, DICOM, and FITS. [Open Source, Public Domain]
- Open source for image processing tasks: http://octave.sourceforge.net/doc/image.html

---

JOURNALS

Elsevier

- Computer Vision and Image Understanding
- Journal of Visual Communication and Image Representation
- Data & Knowledge Engineering
- Image and Vision Computing
- Pattern Recognition
- Pattern Recognition Letters

IEEE

- IEEE Transactions on Circuits and Systems for Video Technology
- IEEE Transactions on Multimedia
- IEEE Transactions on Image Processing
- IEEE Transactions on Medical Imaging
- IEEE Transactions on PAMI

Computer Vision

- *Computer Vision Homepage at CMU*
- Annotated Computer Vision Bibliography from USC IRIS
- *CVonline: The Evolving, Distributed, Non-Proprietary, On-Line Compendium of Computer Vision*

- **3-D for Everyone**

- Red-blue glasses or anaglyph for 3D viewing: http://www.best3dglasses.com/anaglyph.html
- Shutter glasses for 3D viewing: http://www.stereo3d.com/shutter.htm
- 3D cameras: http://www.ptgrey.com/index.asp
- 3D photos at http://www.jessemazer.com/3Dphotos.html
- 3D video sequences can be downloaded at: http://research.microsoft.com/vision/InteractiveVisualMediaGroup/3DVideoDownload/

Public Domain Image Databases

CMU Database