

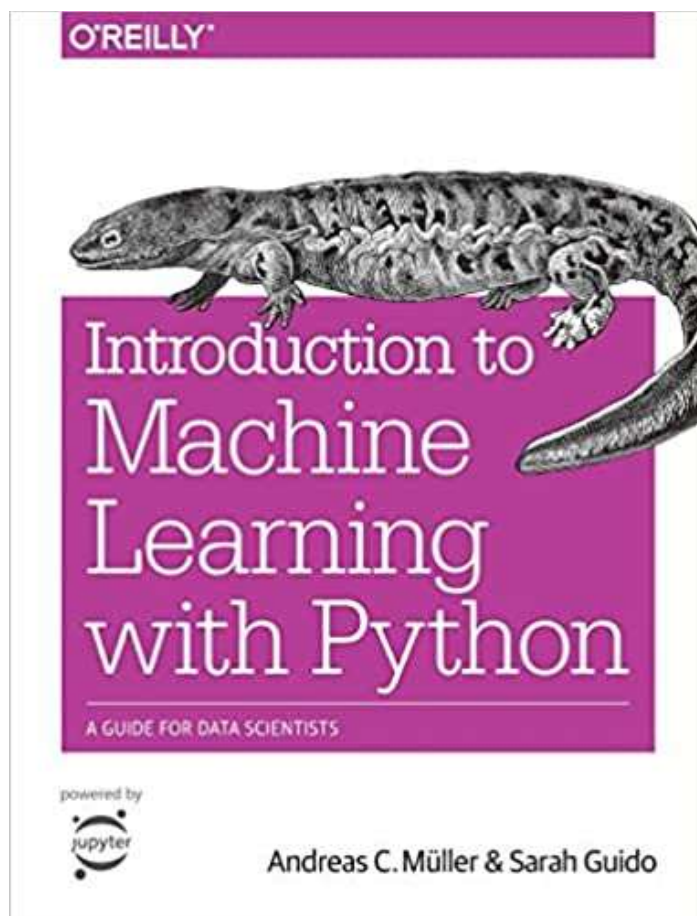
CS-482 MACHINE LEARNING

CS-682 MACHINE LEARNING

PREREQUISITES: CS-102-Computing and Algorithms II for CS-482
CS-601 Programming Introduction for Data Scientists for CS-682

INSTRUCTOR : Prof. Saroja Kanchi

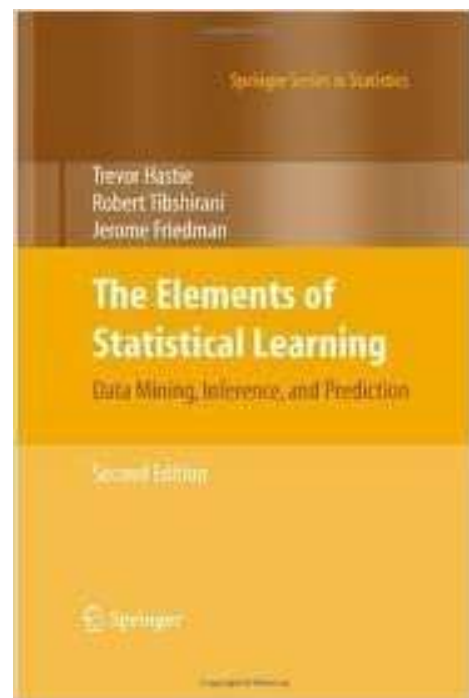
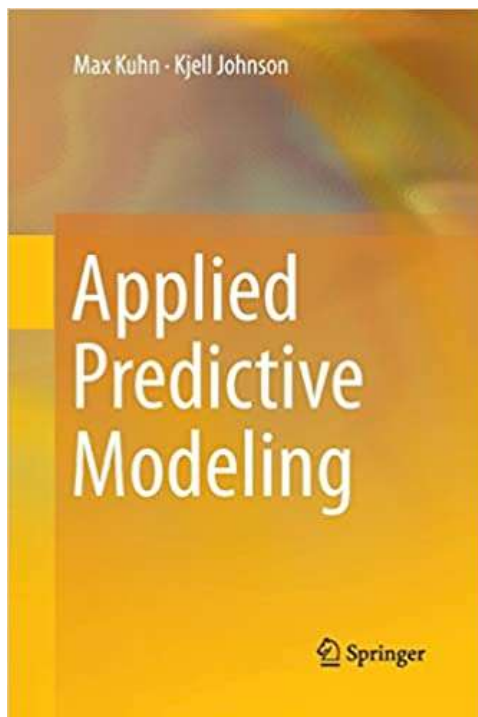
BOOKS : *Required* Introduction to Machine Learning with Python: A Guide for Data Scientists 1st Edition by [Andreas C. Müller](#) (Author), [Sarah Guido](#) (Author) ISBN: 9781449369415



BOOKS : *Recommended*

Applied Predictive Modeling Softcover reprint of the original 1st ed. 2013 Edition by [Max Kuhn](#) (Author), [Kjell Johnson](#) (Author)

The Elements of Statistical Learning: Data Mining, Inference, and Prediction (Springer Series in Statistics) 2nd ed. 2009. Corr. 3rd printing 5th Printing. edition Hardcover – January 1, 2008 by [Trevor Hastie](#) (Author)



EMAIL: skanchi@kettering.edu.

Please email any quick questions between class meetings. For longer discussions, please stop by during student hours.

WITHDRAW DATE: You may withdraw from the course with no grade assigned to the course up to Friday of Week 7. (with no refund) .

COURSE DESCRIPTION: An introduction to machine learning with application to big data. Topics include: supervised learning including generative, discriminative learning, parametric and non-parametric learning, neural networks, support vector machines; unsupervised learning including clustering, dimensionality reduction, kernel methods; learning theory bias/variance tradeoffs; VC theory; large margins; reinforcement learning. Applications of machine learning to big data.

COURSE LEARNING OBJECTIVES:

Each student who receives credit for CS-482 will have demonstrated the ability to do each of the following tasks:

1. To be able to explain the basic building blocks and general principles of machine learning algorithms
2. To be able to compare and contrast the strength and weaknesses of machine learning algorithms
3. To be able to demonstrate the application of machine learning algorithms to real world data and evaluate their performance

WEEKLY COVERAGE OF MATERIAL (TENTATIVE)

Week Number	Topics Covered	Sections Covered/Reading Assignment	Assignments/Tests
Week 1 Day 1	Ch1- Intro to ML and Tools	Syllabus, IMLP Ch1, APM Ch1-2	
Week 1 Day 2	Ch1 -Intro to ML and Tools	IMLP Ch1, APM Ch1-2 Intro	Install Software in Ch1 slides, A1 Assigned
Week 2 Day 1	Ch2- kNN and Cross Validation	IMLP 2.1-2.3.2 APM 4.1-4.3 IMLP 5.1-5.2	
Week 2 Day 2	Ch2- kNN and Cross Validation	IMLP 2.1-2.3.2 APM 4.1-4.3 IMLP 5.1-5.2	
Week 3 Day 1	Ch3-Linear Models and Evaluation Metrics	IMLP 2.3.3 IMLP 5.3	

		APM Ch 5, 6 APM Ch 11,12	
Week 3 Day 2	Ch3-Linear Models and Evaluation Metrics	IMLP 2.3.3 IMLP 5.3 APM Ch 5, 6 APM Ch 11,12	A1 Due- Friday of Week 3, A2 Assigned
Week 4 Day 1	Ch4-Data Preprocessing	IMLP Ch3.3 IMLP Ch 4.1-4.4	
Week 4 Day 2	Ch4- Data Preprocessing	IMLP Ch3.3 IMLP Ch 4.1-4.4	Test1- Ch1Ch2Ch3
Week 5 Day 1	Ch5-Nonlinear Models	APM Ch 7 APM Ch 13 IMLP 2.3.4 IMPLP 2.3.7	
Week 5 Day 2	Ch5-Nonlinear Models	APM Ch 7 APM Ch 13 IMLP 2.3.4 IMPLP 2.3.7	A2 Due Friday of Week 5 A3 Assigned
Week 6 Day 1	Final Project Proposal Presentations or Catchup day	Project Proposal Presentations	
Week 6 Day 2	Project Proposal Presentations	Project Proposal Presentations	Proposal Due
Week 7 Day 1	Ch6-Dimensionality Reduction and Feature Extraction	APM Ch 7 APM Ch 13 IMLP 3.4 IMLP 4.7	
Week 7 Day 2	Ch6-Dimensionality Reduction and Feature Extraction	APM Ch 7 APM Ch 13 IMLP 3.4 IMLP 4.7	Test 2, Ch4, Ch5
Week 8 Day 1	Ch7 Tree Based Learning	APM Ch 8 IMLP 2.3.5 2.3.6	
Week 8 Day 2	No class labor day	No class labor day	A3 Due – Friday of Week 8
Week 9 Day 1	Ch7 Tree Based Learning	APM Ch 8 IMLP 2.3.5 2.3.6	
Week 9 Day 2	Ch8 Working with TextData	Ch 7 IMLP	
Week 10 Day 1	Final Project Work		Test 3: Ch6,Ch7, Ch8
Week 10 Day 2	Final Project work		
Week 11 Day 1	Final Project Presentations		Final Project Presentations

Week 11 Day 2	Final Project Submission		
---------------	--------------------------	--	--

If a class is shortened/cancelled due to unexpected circumstances the material will be caught up in the following class session when the class meets and schedule will be changed for following weeks.

CLASSROOM EXPECTATIONS

1. You are expected to attend ALL classes and participate in the discussions.
2. Refrain from talking to each other during the lectures. If you have a comment about the subject at hand, please share with all of us.
3. Come to classroom on time and do not leave the class in the middle of the lecture.
4. You should keep track of your grade at all times. If you see yourself performing poorly, please take early action to meet and discuss your situation and seek help.
6. **Submission of work not entirely your own, will receive an automatic zero**, with a warning, for the first offense. An automatic failing grade will be assigned for the course for any subsequent offense. This is applicable to all parties involved in the cooperative effort. So please do not help others in assignments or tests. If you help someone to the point where both of you end up with essentially same assignment, then you are also equally guilty as the person(s) that got help from you. Documentation related to any incidence of cheating in assignments or tests will be forwarded to the Dean of Academic Affairs and will be added to the student's file.

EXPECTED BEHAVIORS OF STUDENTS FOR SUCCESSFUL COMPLETION OF THE COURSE

1. Attend all classes.
2. Take notes during the lecture.
3. Pay undivided attention to class (No use of cell phones or other distractions)
4. Follow all written and oral instructions.

5. Complete your reading assignment prior to coming to class.
6. Do your homework (even if it is extra credit)
7. Start programming assignments early.
8. Come in for a personal meeting, if you do not understand the material or need help with assignments.
9. Send email asking any clarification questions.
10. Prepare diligently for midterm and final exam.

GRADING POLICIES

The grades for the course will be split among the following testing mechanisms.

Attendance and Participation- 10%

Tests- 30%

Programming Assignments--- 40%

Final Project 20%

CURVE: Each individual's grade will be adjusted upward (if needed) to make the class average of 85 at the end of the course. Class averages for individual quizzes or assignments or midterms will not be computed or provided. The class average will be computed on the overall grade only at the end of term and necessary points will be added if the average is below 85. Note also that the average is computed for grades over 45% only.

Letter grades will be assigned using the scale below once the curved grade is computed.

90- 100 Excellent (90 – 92 is A- and 93-100 is A)

80 -89 Above average (80 – 82 is B-, 83-86 is B, and 87-89 is B+)

70 – 79 Average (70-72 is C-, 73-76 is C and 77-79 is C+)

60- 69 D+

50-59 D

< 50 is failing or F

ATTENDANCE AND PARTICIPATION: Attendance includes attending classes and active participation in class.

TESTS: There will be three tests during the term. These will be based on material taught in the previous three weeks and homework assigned for the three weeks. Tests will be based on homework questions so it is important to do the homework.

PROGRAMMING ASSIGNMENTS: There will be three programming assignments. Within the first week, please fill in the google shared form indicating who your partner is. If later you decide to work alone, let me know about this and you can then work alone. However, you may not change partners once chosen.

FINAL PROJECT: The final exam will be replaced by a final project of your choosing. This will be a group project that must be done with *groups of three students*. The due dates for various steps of the project will be available in a separate document. The final project presentation will be during Week 11.

GRADING : There will be one week of time used for grading programming assignments and less than a week to grade written exams. Full credit will be given to correct solutions that meet all of the requirements. Partial solutions will receive partial credit. The amount of partial credit is dependent on the rubric used for grading.

You are required to keep the returned graded material in a safe place until a grade is assigned for the course.

LATENESS:

All of the items assignments and term project is due on the announced due date. You are given TWO late COUPONS for the entire term. You may use each coupon with the following privileges.

1. turn in one assignment late by 24 hours,

When you are turning in the assignment on blackboard, let me know that you are using one of your coupons.

2. Note that coupons cannot be used to gain attendance points when you miss a class

Note that if you do not use the coupon for the class, by the end of term, it cannot be refunded for grade!

ABSENCES

You are required to attend all the classes so that you perform well in the course. If you miss a class in the term, please read the missed chapters from the textbook, and, you may meet me during office hours for any clarification.

If you attend very few of the classes in the first four weeks, Academic Service Center will be notified of your absence

If you cannot take the midterm or any of the tests, you should reschedule the test as early as possible and alternate arrangements will be made. You may reschedule the tests only with proper documents from a medical professional. One coupon will be used to make up a test.

CS-682 Requirements:

CS-682 students will be required to answer additional questions in each test and also fulfill additional requirements in each assignment and the final project.

PLAGIARISM CHECK IN PROGRAMMING ASSIGNMENTS AND FINAL PROJECT:

Plagiarism checking software will be used to check for plagiarism in assignments. No code can be copied from the internet w/o giving credit to the webpage where you found it. Code from textbook can be freely used.

UNIVERSITY POLICIES: Please see University Policies document in the Syllabus tab for additional university policies.

Instructions for Submitting Programming Assignments

Follow the following procedures for turning in your assignment. Note that grading rubric includes a portion of the grade for following the submission procedures.

Name all your files and folders using camel case, with first letter of the name is small case. If you are working in a team of two, then both names MUST appear in all files and submissions.

Step1:

Create a folder name <yourLastName(s)#> where yourLastName(s) is your last name (and your teammates). And the # is the assignment number. For the first assignment, use number 1 and for the second assignment, use 2 etc. For example if you last name is Allen you will use allen1 as the folder name. Do not use spaces or any punctuation in naming this folder. Use all small case letters in the name of the folder.

Step 2

Place all the relevant files into this folder including any pdf files, python files etc.

Step 3:

Zip the folder <yourLastName#> to create a compressed file, <yourLastName#>.zip and upload this zipped file. Do NOT ZIP THE FILES, INSTEAD ZIP THE FOLDER.

There are several ways to zip a folder. If you right click on the folder and you see an option to create .zip file, use that option. Or use the following command

```
zip allen1.zip allen1/*
```

This compresses all files in allen1 folder and places in allen1.zip

CHEATING IN PROGRAMMING ASSIGNMENTS

It is expected that you write 100% of the program will be written by you and your teammate (if working in groups). You must not get help from anyone outside the class (other than your teacher), or inside the class. No AI programs such as ChatGPT can be used.

If two or more programs have essentially same code, (just changing variable names and function names will not make the code dissimilar) or code is generated from ChatGPT, the parties involved will be considered to have cheated. If I determine

that cheating has occurred, I will follow university policies regarding handling plagiarism issues.