CSE514 Fall 2019 - Datamining
Administrative Information and Syllabus

Instructor: Professor Weixiong Zhang; email: zhang@cse.wustl.edu; homepage: www.cse.wustl.edu/~zhang; office: Jolley 530.

Prior knowledge required: The knowledge of the following topics is essential to take and do well in this course: Calculus, Data Structure and Algorithms, Probability and Statistics, and Matrix Algebra. Mastering of at least one programming language (e.g., Python, MatLab, C/C++, Java, and R) is also required.

Programming skills and programming environment: This is not a programming course, but you must know how to program (see requirement above). Importantly, you are responsible to have an efficient programming environment setup for the course project. I highly recommend to use Python, Jupyter notebooks and Google’s Colab and other related tools so you can run some programs on Google’s GPU machines for free if you don’t have your own local GPU box. Having access to GPU will help tremendously for the project of the course.

Main objectives of the course: You should expect to walk away from the course the knowledge of the basic concepts and methods of DM (and machine learning), as well as some working knowledge on how a data analytic problem should be approached, i.e., knowledge on what data processing and analysis methods should be adopted in order to solve the problem.

My way of teaching and how you should take the course: I use an old-school fashion of teaching, i.e., a type of chalk talk to go over a subject and write and draw on board the key concepts and ideas, and work through examples. I will also ask questions along the way to inspire you to think and get involved.
I will release my lecture notes for the topics to be discussed in class. You are not required to review the materials - in fact, you are discouraged to preview my lecture notes. Instead, you are strongly encouraged to bring a copy of my lecture notes to the classes so that you may add your own notes and understanding on a topic to make a complete set of notes for yourself. You are required to review the materials we cover in class. I want to highlight that the lecture notes that I provide serve primarily as a guidance to my lectures and help you take notes. As such, the original, incomplete lecture notes may not be an ideal resource for independent study. The combination of my notes and your notes taken during the lectures should be your textbook for the course. While class attendance is not mandatory, you are responsible for all of the content we discuss in class. We will have in-class quizzes based on the previous lectures, which contribute to your final grade. This implies that you are better off to attend the class and participate. In short, the more active and being involved in the course (i.e., attending the course, finishing the assigned reading materials, reviewing after each lecture/topic, preparing well for quizzes/exams, and spending enough time on the course project), the better off you will be with the knowledge you will walk away with and a good grade you expect to earn.

The course is managed online in Canvas, meaning that most teaching materials are distributed there. We will also use an online forum on Piazza for off-hour, online discussion. You will receive an invitation from the system to join the forum.

Textbook, reference and software tool kit: There is no required textbook for this course, mainly because the topics we cover are not in a single book. Nevertheless, the following are good reference books for the course and for learning Datamining/Machine Learning in general:

  Note: This book provides a comprehensive description of the topics and methods in the tool kit WEKA - google to find it online.

  Note: This should have a good coverage of many topics we cover, except deep learning.

- *Deep Learning*, by Ian Goodfellow, Yushua Bengio and Aaron Gourville,
MIT Press.
Note: An electronic version is available online; just google it.

- *Deep Learning with Python*, by Francois Chollet, Manning Shelter Island Publisher.
  Note: This is written by the developer of Keras, and the book is a good source for learning Keras. An electronic version is available online; just google it.

You should try to get the latest versions of the first two books - looking them up on the internet or Amazon.

In addition, many online documents serve well as additional reading materials. I will post the links or pdf documents to relevant topics to help with learning when necessary. Some of these documents are required reading as I won’t have the time to cover all the basic materials in class. Although you may choose not to read these online materials, some of the concepts and problems in quizzes/exams may come from these materials.

**Course work and grading:** Your work involves periodic in-class quizzes (30%), two exams (30%), and a course project with multiple components (40%). One of the lowest quiz score will be excluded from your final grade, so please don’t ask for makeup quiz if you cannot come to a class when we have a quiz scheduled. The topics to be covered in each quiz and the date for the quiz will be announced one week ahead of time to help you prepare. **The course project requires programming.** Also note that there is no final exam for the course, instead we have two exams - the first will be scheduled around the Spring Break and the second in the last class of the course.

In case you have a problem with the grading of any of your work (quiz, exam, and project), you have ten (10) calendar days after we finish grading and return your work to ask for a regrading, if needed. We will never regrade any previous work if it has passed the regrading deadline. **Note that the result of a regrading can go either way** - you may get more points or you may lose more points depending on the problem and your answer(s).

**Collaboration policy:** Collaboration on quiz or exam is NOT allowed, while collaboration on project is encouraged. Serious violation (e.g., plagiarism) will be reported to the dean’s office. Any plagiarism will not be tolerated: you get zero on a quiz/exam if this happen the first time, and
zero for the course if it happens the second time.

Office hours:
Weixiong Zhang, office: Jolley 530, time: Mondays 11am-12pm; Wednesdays 3:30-4:30pm.
TAs: TBA

Main topics (which may be adjusted or revised to add or remove topics as needed)

1. Introduction
   - Problems that DM attempt to solve
   - The process and elements of a DM system.
   - Characteristics of data - big data means big on size, but what size?
   - Supervised vs. unsupervised learning.
   - Datamining vs. Machine Learning vs. Math vs. Computer Science
   - Similarity measure - similarity vs. distance

2. Supervised learning - Regression and classification
   - Regression
   - k-nearest neighbors
   - Artificial neural networks (ANNs)
     - Optimization: Gradient descent and Backpropagation
   - Autoencoder (it’s here for a good reason)
   - Decision trees, random forest and bagging (reading based)
   - Kernel methods and support vector machines (SVMs)
   - Bayesian statistical learning, naive Bayesian, and discriminative
methods

3. Performance measure
   - Elements of performance measure and confusion matrix
   - What to compare, algorithms or problem instances?
   - Receiver Operating Characteristic (ROC) curve

4. Unsupervised learning - Clustering
   - Hierarchical clustering - Agglomerative vs. divisive (reading based)
   - k-means and the EM algorithm
   - Probabilistic model-based clustering

5. Pattern discovery through dimension reduction
   - Curse of dimensionality - The problem
   - Feature selection (reading based)
   - Principle component analysis (PCA)
   - Singular value decomposition (SVD)
   - Nonnegative matrix factorization (NMF) (reading based)
   - Identification of modular structures of network
   - Spectral clustering
   - Spectral graph analysis (optional)

6. Deep learning
   - ANNs as a general model vs deep learning as special architectures
   - Autoencoder (and PCA, see earlier)
   - Convolutional NNs (CNNs)
   - Insights into deep neural networks

7. Association rule mining
   - Frequent set and Apriori algorithm
   - Generation and selection of rules